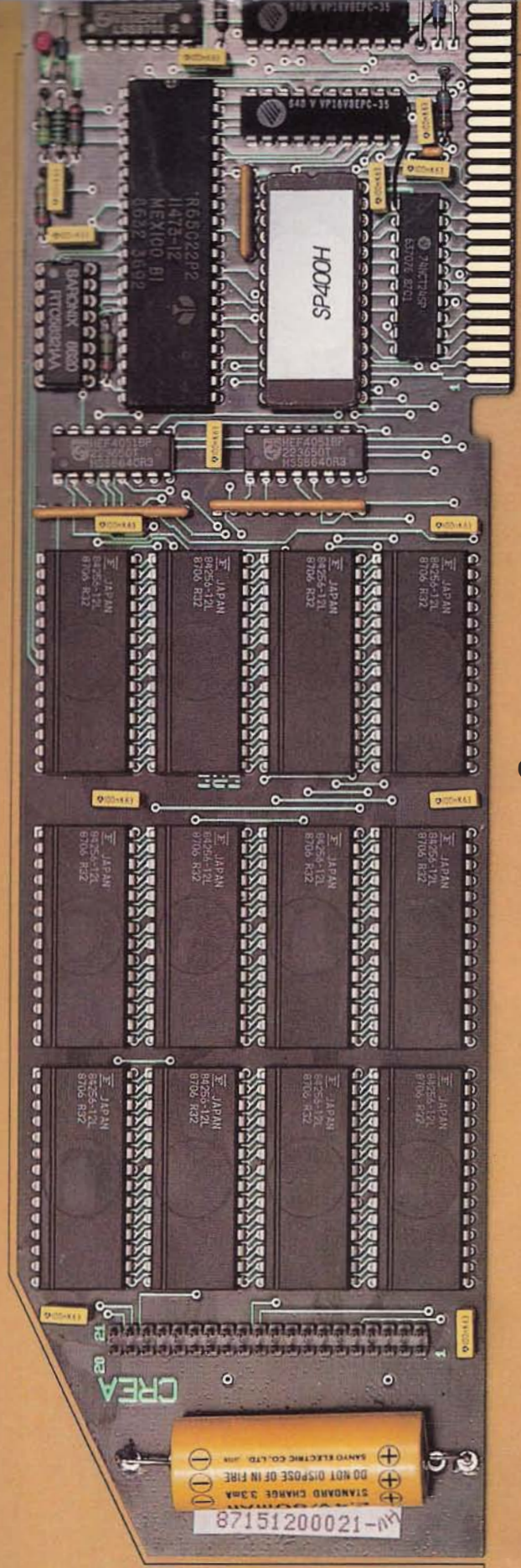
La revue francophone indépendante pour les utilisateurs des Apple][+, //e, //e+, //c, IIGS™ et Macintosh™

pom's

Ouverture sur le monde...

ISSN 0294-6068
M 2366 - 33 - 45,00 F
3792366045005 00330



L'anti-disque

Speedisk™, la RAM Card à mémoire permanente

Rapidité Temps d'accès à l'Information : 0,2 ms (100 fois plus rapide que les disques durs...).
Par exemple :
démarrage sur Basic.System en 3 s,
AppleWriter disponible en à peine 1 s.

Fiabilité Constituée de circuit CMOS à très faible consommation, Speedisk™ est aussi fiable que l'ordinateur lui-même. Elle est insensible à l'environnement.

Capacité Speedisk™ est proposée en quatre versions :
1 Mo (1 048 576 octets)
384Ko extensible à 1Mo
avec horloge compatible ProDOS (pour les Apple)(+, //e) ou sans (IIGS)

Compatibilité 100% compatible avec ProDOS (c'est un volume), Speedisk™ fonctionne sur Apple)(+, sur Apple //e et sur Apple IIGS.

Prix Lecteur de Pom's, vous bénéficiez d'une remise de 10 % :

SP400		
384Ko	~~5 990,00~~	**3 591,00**
SP1000		
1 Mo	~~5 990,00~~	**5 391,00**
SP400H		
384Ko horloge	~~4 580,00~~	**4 122,00**
SP1000H		
1 Mo horloge	~~6 580,00~~	**5 922,00**

Garantie Speedisk™ est une fabrication française garantie un an par échange de la carte.

Banc d'essai dans la revue Pom's n° 31

Vente par correspondance, documentation, renseignements
Logma S.A.
12, rue d'Anjou
78000 Versailles
Tél : (1) 39 51 24 43

Speedisk™ est une marque déposée de Thot Informatique® – France

Speedisk™

Je désire recevoir – sans engagement – votre documentation sur les cartes Speedisk™

Nom.....
Adresse.....
.....
.....

Numéro 33
novembre-décembre 1987

Sondage Pom's — Page 4

Éditorial — Hervé Thiriez — Page 5

La date avec AppleWriter — Lyda Pátková — Page 6

Graphisme sur Minitel — Patrice Neveu — Page 11

Raccourcir WPL — Page 15

GS & QuickDraw II — LE Pham Hiep — Page 17

Heure & AppleWriter — Lyda Pátková — Page 24

Commande externe : ONLINE — Pascal Cantot — Page 25

Impression avec AppleWriter — Page 28

Cats & Mouse — Pascal Duboin — Page 29

En vitrine — Page 40

Graphisme & Minitel — Jean-Luc Bazanegue — Page 41

Essai Macintosh : SideKick version 2 — Hervé Thiriez — Page 47

Recherche d'applications — Georges Zwingelstein — Page 49

Booter dans un dossier ? NEWBOOT — Patrick Guyou — Page 55

Disquettes Pom's 33 — Page 57

Essai : SuperMacroWorks — Damien Nould — Page 62

AppleWorks & LaserJet+ — Bruno Ribard — Page 63

MultiScribe 2.0 — Bernard Toméno — Page 65

Apple // for ever — Jean-Yves Bourdin — Page 66

sur CalvaCom — Page 69

Bibliographie — Alexandre Duback — Page 72

Mots croisés — Joëlle Piard — Page 73

Les annonceurs ; Apple : page 38 & 39 ; La Commande Électronique : page 76 ; LOGMA S.A. : page 2.

Éditions MEV - 12, rue d'Anjou - 78000 Versailles. Tél. : (1) 39 51 24 43. Directeur de la publication : Hervé Thiriez

Sondage

Ce bref questionnaire est pour nous une précieuse source de renseignements : ses précédentes apparitions sont à l'origine des disquettes DOS/ProDOS, des disquettes 3,5", des programmes de communication. Il nous permettra de suivre au mieux l'évolution de vos besoins et celle de votre matériel. Il constituera également un moyen de gagner l'un des 10 abonnements avec disquettes...

Vous avez :
- de 15 ans ..... O
- de 15 à 19 ans ..... O
- de 20 à 24 ans ..... O
- de 25 à 34 ans ..... O
- de 35 à 45 ans ..... O
- + de 45 ans ..... O

Sexe
- Masculin ..... O
- Féminin ..... O

Les langages que vous pratiquez aisément :
- Basic ..... O
- Pascal ..... O
- Forth ..... O
- Logo ..... O
- Asm Apple // ..... O
- Asm Mac ..... O
- Autres
- .....

Votre niveau en programmation :
- Utilisateur ..... O
- Débutant ..... O
- Intermédiaire ..... O
- Averti ..... O
- Haut niveau ..... O

Chaque semaine, vous êtes devant votre micro :
- de 2 heures ..... O
- de 3 à 8 heures ..... O
- de 9 à 16 heures ..... O
- + de 16 heures ..... O

Votre ordinateur :
- Apple][+ ..... O
- Apple //e ..... O
- Apple //e+ ..... O
- Apple //c ..... O
- Apple IIGS ..... O
- Apple /// ..... O
- Compatible ..... O
- Macintosh 128 ..... O
- Macintosh 512 ..... O
- Macintosh Plus ..... O
- Lisa/Mac XL ..... O
- Mac II/SE ..... O

Votre imprimante :
- ImageWriter ..... O
- ImageWriter 2 ..... O
- Apple DMP ..... O
- LaserWriter ..... O
- Epson ..... O
- Seïkosha ..... O
- Mannesman ..... O
- Centronics ..... O
- OKI ..... O
- Scribe ..... O
- Autre
- .....

Extensions, périphériques :
- Carte 80 colonnes ... O
- 80 col. étendue ..... O
- Carte Z80 CP/M .... O
- Carte souris ..... O
- Disque dur ..... O
- Lecteur 140Ko ..... O
- Lecteur 800Ko ..... O
- Table traçante ..... O
- Minitel ..... O
- Modem ..... O
- Extension mémoire O
- Speedisk ..... O
- Autre :
- .....

Prochainement, vous allez acquérir :
- Apple // ..... O
- Macintosh ..... O
- Lecteurs ..... O
- Minitel ..... O
- Modem ..... O

CalvaCom :
- Vous êtes abonné ... O

Depuis :
- 6 mois ou plus ..... O
- Moins de 6 mois ..... O

Combien de numéros de la revue Pom's avez-vous lu ? ..... ___

De 5 (très bon) à 0 (très mauvais), votre opinion sur :
- Niveau général .... ___
- Présentation ..... ___
- Intérêt des programmes ..... ___
- Clarté des articles ..... ___
- Intérêt pédagogique ..... ___
- Micro-Infos ..... ___
- Disquettes ..... ___

Ce que vous préférez :
- .....
- .....
- .....
- .....
- .....

Ce que vous détestez :
- .....
- .....
- .....
- .....
- .....
- .....

Ce qu'il faut changer :
- .....
- .....
- .....
- .....
- .....

Ce qu'il faut développer :
- .....
- .....
- .....
- .....

Vos autres revues informatiques :
- .....
- .....
- .....
- .....

Vos remarques :
- .....
- .....
- .....
- .....
- .....

FACULTATIF

Nom
.....
Adresse
.....
.....
.....
N° CalvaCom
.....

Abonné ..... O

Éditorial

Ont collaboré à ce numéro
Alexandre Avrane – Jean-Luc Bazanegue
Alain Bohec – Jean-Yves Bourdin
Pascal Cantot – Alexandre Duback
Pascal Duboin – Patrick Guyou
Le Pham Hiep – Olivier Herz
Gérard Michel – Patrice Neveu
Damien Nould – Lyda Pátková
Christian Piard – Joëlle Piard
Bruno Ribard – Georges Zwingelstein
Hervé Thiriez – Bernard Toméno

Directeur de la publication
rédacteur en chef
Hervé Thiriez

Rédacteurs
Alexandre Avrane – Olivier Herz

Siège social
Éditions MEV
12, rue d'Anjou
78000 Versailles
☎ (1) 39.51.24.43

Publicité
Éditions MEV

Diffusion
N.M.P.P.

Impression
Berger-Levrault
18, rue des Glacis
54000 Nancy
☎ 83.35.61.44

Photo de couverture
CP & JLB

Illustrations 'Apple // for ever'
Jean-Yves Bourdin

Photogravure
Graphotec
21, chemin de la Tour
92350 Le Plessis-Robinson
☎ (1) 46.30.44.49

Pom's est une revue indépendante non rattachée à Apple Computer, Inc. ni à Apple Computer France S.A.R.L.
Apple, le logo Apple, Mac et le logo Macintosh sont des marques déposées d'Apple Computer, Inc.
IBM est une marque déposée de International Business Machine.
PC et AT sont des marques déposées de la Société IBM.

Pom's est entré dans sa septième année !
Nous avons vu beaucoup d'eau passer sous les ponts. Nous avons vu naître et mourir l'Apple III, Golden, Infomag. Maintenant, c'est l'Apple IIc dont la vente est interrompue. La gamme Macintosh est apparue et porte à présent très haut le flambeau, ayant même réussi la percée dans les grands comptes ce que les autres modèles de la marque n'étaient jamais parvenus à faire ; il entraîne dans sa lancée l'Apple IIGS qui devient incontestablement un 'grand' de la gamme. Apple Expo, chaque année une réussite, nous rassure sur l'avenir de nos chers micros : le fruit du travail des développeurs aussi bien dans la création de programmes que dans le 'hard' semble bien sans limite.

Dans cette optique, nous continuons à faire évoluer Pom's pour vous apporter une qualité de service grandissante. C'est aujourd'hui la possibilité, par l'intermédiaire de notre boîte aux lettres CalvaCom, de passer des commandes télématiques et de nous soumettre vos problèmes techniques dans la mise en œuvre des programmes de la revue : il nous arrive de répondre à des heures fort tardives et même durant le week-end... Il nous faut hélas, lors d'affluence, privilégier nos abonnés dans notre assistance technique : abonnez-vous !

S'abonner, c'est — outre une priorité dans l'aide que nous vous apportons — recevoir Pom's avant la mise en kiosque, c'est passer gratuitement dans votre revue des petites annonces, c'est également recevoir par tirage au sort des cadeaux-surprise de valeur : logiciels, documents techniques (nouveaux abonnements souscrits à partir du 30 novembre 87). C'est aussi un rapport plus étroit avec Pom's, un plus grand poids dans vos suggestions.

Idées, nouvelles orientations ? Notre sondage ci-contre nous permet de mieux vous connaître, c'est une sorte de trait d'union.

Notre ami Jean-Michel Gourévitch est en route vers de nouvelles aventures et c'est désormais Jean-Yves Bourdin qui assurera la chronique des nouvelles sur Apple //. À cette occasion nous vous proposons une nouvelle formule, "Apple // for ever", composée de huit rubriques. Là encore, c'est à vous de dire ce que vous pensez de cette innovation et aussi à vous d'y contribuer : envoyez-nous nouvelles, informations, 'patches'...

Le contenu de ce numéro satisfera certainement les besoins de chacun ; en couverture, l'un des résultats de nos recherches dans le domaine de la communication : le couple micro-ordinateur/Minitel est doué d'un potentiel peu commun...

Hervé Thiriez

La date avec AppleWriter

Lyda Pátková

La carte Speedisk, grâce à l'extrême rapidité d'accès aux fichiers, permet d'envisager de nouvelles applications WPL – langage de programmation d'AppleWriter –, applications qui resteraient lentes d'exécution sur les supports traditionnels ; les programmes présentés ici demeurent toutefois utilisables sur tous types de supports.

Nous vous proposons deux programmes :

- le premier permet de disposer de la date dans un fichier à charger à tous moments par une commande glossaire par exemple ;
- le second montre un exemple d'automatisation du courrier répétitif habituel, devis, commande, relance... Cet exemple utilise la date et gère une numérotation de documents.

Bien sûr, pour que tout cela fonctionne, il faut disposer de la date. Il y a trois méthodes :

- doter son Apple d'une carte horloge ;
- y installer une Speedisk avec option horloge ;
- disposer d'un programme de boot qui demande la date et la met à jour dans les octets $BF90/$BF91 de ProDOS. REBOOT.SYSTEM disponible dans la bibliothèque de CalvaCom convient fort bien.

La date

Le premier programme WPL, qui se contente de vous offrir la date à la française (03 novembre 1987), s'exécute en faisant simplement :

```
<ctrl-P>DO STARTUP.D
```

Avant cet ordre, vous aurez mis sur votre disquette (ou plutôt volume) les fichiers STARTUP.D, INIDATE.D (le fichier intermédiaire VARIDATE.D se crée automatiquement) et DATE.

Un instant plus tard, le fichier DATE créé sur le volume contient la date du jour : pour l'utiliser faire simplement :

```
<ctrl-L>DATE
```

Mieux : installer la commande glossaire

```
D<ctrl-L>DATE>
```

ainsi, la date sera sous la pomme : ⌘D.

Concrètement, le fonctionnement se résume ainsi :

STARTUP.D

C'est le premier programme à s'exécuter. Il sauvegarde un fichier vide dans le catalogue courant puis charge ce catalogue en mémoire texte. La date en cours est la date de modification du fichier "VIDE" sous la forme : mm/jj/aa. STARTUP.D isole donc cette date en mémoire texte et la transforme en un programme WPL sauvegardé sous le nom de VARIDATE.D et exécuté à son tour par la dernière commande de STARTUP.D.

VARIDATE.D

Ce programme n'a qu'une fonction : il place dans les variables $B le mois, $C le jour, $D l'année, ce qui va permettre de les traiter à notre convenance. Cela fait, VARIDATE.D lance l'exécution de INIDATE.D, 3ème programme WPL (heureusement que Speedisk met un tigre dans AppleWriter...).

INIDATE.D

Il était peu utile de garder la date sous la forme *01/01/1987*, INIDATE.N peaufine le travail en la présentant sous la forme *01 janvier 1987* et en la sauvegardant dans le fichier texte "DATE" (remarquer le PGO Variable bien utile dans le programme WPL). En fin d'exécution, DATE est disponible.

Remarque : si aucun de vos programmes WPL n'utilise l'une des quatre variables disponibles ($D, par exemple), cet ensemble pourrait simplifié. La date mise en forme serait placée non plus dans un fichier mais dans la variable $D, INIDATE se terminant alors de la façon suivante :

...

```
12 pas décembre =$C
 pas $B$C $D=$D
```

$D resterait alors utilisable autant de fois que nécessaire au sein d'un programme WPL par la seule fonction du type :

```
F/(DATE)/$D/
O?
```

STARTUP.D, VARIDATE.D et INIDATE.D ne sont exécutés qu'une fois, à la mise en route d'AppleWriter. Si STARTUP.D est renommé STARTUP, vous bénéficierez d'une exécution automatique lors du démarrage.

Un exemple complet

L'exemple d'utilisation qui suit inclut, en plus de l'usage de la date tel que décrit, une fonction de numérotation automatique des documents retenant la forme : numéro d'ordre numéro du mois (6709, 6809, 6910, 7010, etc). Il conduit à la réalisation semi-automatique de commandes ou de relances.

À cette fin le programme STARTUP.N remplace le précédent STARTUP.D. STARTUP.N crée un programme VARIDATE.N complété par la mise à jour du numéro du mois dans le fichier NUMERO.

STARTUP.N

La comparaison des deux startups et des programmes VARIDATE respectifs qu'ils créent, permet de bien comprendre le processus de transformation de la date, initialement isolée du catalogue, en un programme WPL.

Remarquer l'usage dans la 18ème ligne de §C au lieu de $C. S'il était écrit $C, l'exécution de la fonction de remplacement mettrait la valeur courante de la variable $C au lieu du nom de variable $C. Reste ensuite à ajouter la commande de remplacement de § par $ (c.f. fin du programme).

Est ajouté aussi le chargement du

glossaire (QEGLOSSAIRE) dont la seule commande prévue permet revenir à tout moment au programme CHOIX. Cette commande est donc à ajouter à votre glossaire habituel.

VARIDATE.N

VARIDATE.N contient maintenant la fonction de mise à jour du numéro du mois dans le fichier NUMERO. Il lance l'exécution de INIDATE.N

INIDATE.N

Sans changement, lance l'exécution de CHOIX.

CHOIX

Exécuté pour la première fois automatiquement par l'intermédiaire de INIDATE.N, il doit être le passage obligé de toute nouvelle application d'où la commande incluse dans le glossaire.

Pour limiter cet exemple, seules les options 1 et 3 sont utilisées. L'option 1 permet de préparer une commande et l'option 3 une relance.

Les programmes WPL RELANCE et COMMANDE sont simplifiés. Ces programmes doivent charger les fichiers de tabulation et d'impression appropriés. Ce programme contrôle la validité du choix à l'aide, d'une part de la fonction de comparaison : contrôle de Escape, et d'autre part d'une boucle numérique limitant la valeur choisi entre 0 et 4 : 0<CHOIX>4.

GLOSSAIRE

Le glossaire ne contient que la commande <ctrl-C> <ctrl-P> doCHOIX> (remarquer que > après DOCHOIX signifie RETURN mais certaines versions d'AppleWriter exigent à la place le signe §) qui permet par la commande : ⌘ctrl-C d'appeler CHOIX après chaque application.

COMMANDE (RELANCE)

Ce petit programme WPL permet de placer dans une fiche standard (FICHCOMMANDE) préparée et mise en page à votre convenance, les données variables mais systématiques qui lui sont liées. Pour ce qui nous occupe présentement, il s'agit de la date par les instructions :

```
PLSDATE=$A
...
F/(DATE)/$A/A
```

et du numéro par la même méthode :

```
PLSNUMERO=$C
...
F/(NUMERO)/$C/a
```

Le fichier NUMERO est mis à jour (incrémentation de 100 de sorte que 6709 par exemple devient 6809) et sauvegardé pour l'application suivante.

La lettre de commande créée est sauvegardée sous le nom :

```
C.nom de fichier numéro
```

d'où la commande WPL :

```
SC.$B$C<H><H>
```

où <H><H> permet de supprimer le numéro du mois : si $C = 6709, $C<H><H> = 67.

Le fichier ainsi créé permet de faire la sauvegarde finale par un simple <ctrl-S>=.

Utilisation

Pour essayer l'exemple ci-après proposé il vous faut dans le catalogue courant :

1 *STARTUP.N*
à exécuter en premier
2 *INIDATE.N*
3 *CHOIX*
4 *COMMANDE*
5 *RELANCE*
6 *FICHCOMMANDE*
7 *FICHRELANCE*
8 *NUMERO*
qui contient uniquement le numéro à incrémenter
9 *GLOSSAIRE*
10 *DATE*

Après chaque application on revient directement au menu de CHOIX par la commande glossaire ⌘ctrl-C.

Remarque : pour la simplicité de la présentation, cet ensemble n'utilise aucun sous-catalogue. Il est pourtant préférable de prévoir le classement des fichiers et programmes dans des sous-catalogues spécifiques. En particulier, pour le fichier VIDE, l'idéal est qu'un catalogue lui soit réservé afin que la fonction de recherche de STARTUP soit rapide. Si tel n'est pas le cas il est souhaitable de placer VIDE au début du catalogue courant car STARTUP cherche non pas le fichier s'appelant VIDE mais le premier fichier de longueur nulle !

Fichier STARTUP.D

•no	*Efface mémoire texte*
•<ESC><ESC>	*Efface ligne status*
•PND	*Supprime affichage*
•PPRun•instant•!	*Affiche le message*
•SVIDE	*Sauvegarde un fichier vide*
•0	*Accepte son remplacement*
•OA£	*Charge le catalogue*
•?	*Quitte les cdes ProDOS*
•B	*Va au début catalogue*
•F*•--:--••••••0*	*Positionne le curseur*
•?	*Quitte la recherche*
•D	*Change le sens (<)*
•W	*Charge la date dans le buffer*
•NO	*Efface mémoire texte*
•D	*Change le sens (>)*
•X	*Date en mémoire texte*
•D	*Change le sens*
•F*•*=$D,•NO,•PDOINIDATE.D*	
•O?	*Transforme*
•F!/!£•PAS19!	*la date*
•O?	*en un*
•F!/!£•PAS!	*programme*
•O?	*WPL*
•B	
•F!•!•PAS!	
•O?	
•F*,*=$C,*	
•O?	
•F*,*=$B,*	
•O?	
•SVARIDATE.D	*Le sauvegarde...*
•O	
•pdoVARIDATE.D	*et l'exécute*

][+
//e
//e+
//c
][gs

Fichier DATE

03•novembre•1987	*pas de retour chariot après*

Fichier RELANCE

```
•PND
•PPR<CTRL-L>
•NO
•PLSNUMERO•=$C
•PSX•$C
•PSX•+100
•PAS(X)=$D
•LNUMERO
•F/$C/$D/A
•S=
•PLSDATE=$A
•PPR•Numéro•de•document•••••••••••••••••••••••••$C
•PPR•Date••••••••••••••••••••••••••••••••••••••••$A
•PPR
•NO
•LFICHRELANCE
•BF/(DATE)/$A/
•O?F/(NUMERO)/$C/
•O?
APP•PPR•(1)•Monsieur••(2)•Madame••(3)•Mada
      ¬me,•Monsieur•
•PPR•RETURN•pour•"Messieurs"•(Ets,•SARL,•Etc.)
•PIN••••••••••••••••••••••••••••••••••••••••••••$A
•PCS/$A//
•PGOS
•PSX$A
•PCS/(X)/0/
•PGOAPP
•PSY0
BOUCLE•PSY+1
•PCS/(Y)/4/
•PGOAPP
•PSX-1
•PGOBOUCLE
•PGO$A•
1•F/(APPEL)/Monsieur/a
•PGO•NO
2•F/(APPEL)/Madame/a
•PGO•NO
3•F/(APPEL)/Madame,•Monsieur/A
•PGO•NO
S•F/(APPEL)//
•O?F/(APPEL)/Messieurs/a
NO•PPR•Nom•du•fichier•de•sauvegarde••••••••
•PIN••••••••••••••••sans•signe•ni•espace••••••$B
•PCS/$B//
•PGONO
•PPR•Nom•complet•du•destinataire•(return•=••$B)
•PIN••••••••••••••••••••••••••••••••••••••••••••$D
•PCS/$D//
•PAS$B=$D
•PIN•Prénom••••••••••••••••••••••••••••••••••••$a
•F/(PRENOM)/$a/
•O?F/(NOM)/$D/
•O?PIN•Adresse••••••••••••••••••••••••••••••••••$A
•F/(ADR)/$A/
•O?
A•PIN•Date•de•la•facture••••••../../.••••••••$a
•PCS/$A//
•PGOA
•F!(DATE)!$A!
•O?
F•PIN•Numéro•de•la•facture••••..../..••••••••$A
•PCS/$A//
•PGOF
•F!(NUMERO)!$A!
•O?B
C•PIN•Concernant.........••••••••••••••••••••$a
•F/(CONCERNANT)/•concernant•$a/
•O?
P•PIN•Montant•TTC••(nombre•seul)••••••••••••$a
•PCS/$a//
•PGOP
•F/(MONTANT)/$A/
•O?SR.$B$C<CTRL-H><CTRL-H>
```

Fichier INIDATE.D

```
••NO                    Efface page texte
••PGO$C                 GOTO variable : $C est
00•PGOERR               le n° du mois.
01•pas•janvier=$c
•••pgod                 Changement du n° de mois
02•pas•février=$c       en nom de mois
•••pgod
03•pas•mars=$c
•••pgod
04•pas•avril=$c
•••pgod
05•pas•mai=$c
•••pgod
06•pas•juin=$c
•••pgod
07•pas•juillet=$c
•••pgod
08•pas•aout=$c
•••pgod
09•pas•septembre=$c
•••pgod
10•pas•octobre=$c
•••pgod
11•pas•novembre=$c
•••pgod
12•pas•décembre=$c
d•LDATE                 Charge fichier date
•F*---•+•----*$B$C•$D*
                        En remplace le contenu
•O?
•S=                     Le sauvegarde
•NO
•PPR la date $B$C $D est dis
      ¬ponible dans le fichier "DATE"
•PPR
•PPR voir l'exemple d'utili
      ¬sation avec STARTUP.N
•PPR
•PIN taper une touche :
•POT                    Fin
ERR•PPR<CTRL-L>
•PPR vous n'avez pas de carte h
      ¬orloge ou bien
•PPR vous n'êtes pas passé par R
      -¬EBOOT.SYSTEM
•PPR votre fichier 'vide' n'a do
      ¬nc pas de date de modification
•PPR ce programme n'est pas utilisable.
•PIN
•NO
```

Fichier COMMANDE

```
•PND
•PPR<CTRL-L>
•NO
•PLSNUMERO•=$C
•PSX•$C
•PSX•+100
•PAS(X)=$B
•LNUMERO
•F/$C/$D/A
•S=
•PLSDATE=$A
•PPR•Numéro•de•document•••••••••••••••••••:••$C
•PPR•Date••••••••••••••••••••••••••••••••••:••$A
•PPR
•NO
•LFICHCOMMANDE
•F/(NUMERO)/$C/a
•F/(DATE)/$A/a
NO•PPR•Nom•du•fichier•de•sauvegarde••••••••
•PIN••••••••••••••••sans•signe•ni•espace•:••=$B
•PCS/$B//
•PGONO
•PPR
•PPR•Nom•complet•du•destinataire•(return•=•"$B)
•PIN••••••••••••••••••••••••••••••••••••••:••=$D
•PCS/$D//
•PAS$B=$D
•F/(NOM)/$D/A
•PIN•Adresse••••••••••••••••••••••••••••••••:••=$A
•F/(ADR)/$A/A
•F/Ref/
•?
•JJJI
•SC.$B$C<CTRL-H><CTRL-H>
```

Fichier INIDATE.N

```
••NO
••PGO$C
00•PGOERR
01•pas•janvier=$c
•••pgod
02•pas•février=$c
•••pgod
03•pas•mars=$c
•••pgod
04•pas•avril=$c
•••pgod
05•pas•mai=$c
•••pgod
06•pas•juin=$c
•••pgod
07•pas•juillet=$c
•••pgod
08•pas•aout=$c
•••pgod
09•pas•septembre=$c
•••pgod
10•pas•octobre=$c
•••pgod
11•pas•novembre=$c
•••pgod
12•pas•décembre=$c
d•LDATE
•F*--•+•----*$B$C•$D*
•O?
•S=
•PDOchoix
ERR•PPR<CTRL-L>
•PPR vous n'avez pas de
      ¬carte horloge ou bien
•PPR vous n'êtes pas passé
      ¬par REBOOT.SYSTEM
•PPR votre fichier 'vide' n'
      ¬a donc pas de date d
      ¬e modification
•PPR ce programme n'est pas u
      ¬tilisable.
•PIN
•NO
```

Fichier GLOSSAIRE

```
<CTRL-C><CTRL-P>DOCHOIX>
```

Fichier VARIDATE.D

•PAS11=$C	*Met mois dans $C*
•PAS03=$B	*Met jour dans $B*
•PAS1987=$D	*Met année dans $D*
•NO	
•PDOINIDATE.D	*Exécute Inidate.d*

Concernant les listings

Dans les programmes WPL listés ici :

les '•' symbolisent les espaces,

les caractères qui suivent le signe '¬' sont la suite de la ligne précédente,

<ctrl-L> symbolise le caractère contrôle-L que l'on saisi dans Apple-Writer avec la séquence clavier : ctrl-V ctrl-L ctrl-V ;

Idem pour <esc> à saisir ainsi : ctrl-V touche escape ctrl-V.

Fichier CHOIX

```
•<ESC><ESC>
•PND
OPTION•PPR<CTRL-L>
•PPR
•PPR
•PPR••••••••••••1••commande
•PPR
•PPR••••••••••••2••lettre
•PPR
•PPR••••••••••••3••relance•
•PPR
•PPR•••••••••<Esc>••quitter
•PPR
•PPR
•PIN••••••••••••••••••••••
      ¬•••choisir•une•opti•
      ¬n•1•à•3•:•=$B
•PCS/$B/<ESC>/
•PGOQUIT
•PSX$B
•PCS/(X)/0/
•PGOOPTION
•PSY0
BOUCLE•PSY+1
•PCS/(Y)/4/
•PGOOPTION
•PSX-1
•PGOBOUCLE
•NO
•PGO$B
1•PDOCOMMANDE
2•PPR<ESC>••••••Choisir•1•ou
      ¬•3•pour•un•essai
•PPR
•PIN••••••••••••••••••••••
      ¬••••taper•une•touche•:
•PGOOPTION
3•PDORELANCE
QUIT•<ESC>
•NO
```

ProDOS

Fichier STARTUP.N

```
•NO
•<ESC><ESC>
•PND
•PPRun•instant•!
•SVIDE
•●
•OA£
•?
•B
•F*•--:--•••••••0*
•?
•D
•W
•NO
•D
•X
•D
•F*• *=$D, •NO, •LNUMERO, •HH, •F<?
      ¬?<$C<, •O?, •S=, •PDOINID
      -¬ATE.N*
•O?
•F!/!£•PAS19!
•O?
•F!/!£.PAS!
•O?
•B
•F!•!•PAS!
•O?
•F*, *=$C, *
•O?
•F*, *=$B, *
•O?
•F<$<$<A
•SVARIDATE.N
•●
•QEGLOSSAIRE
•pdoVARIDATE.N
```

Fichier VARIDATE.N

```
•PAS11=$C
•PAS03=$B
•PAS1987=$D
•NO
•LNUMERO
•HH
•F<??<$C<
•O?
•S=
•PDOINIDATE.N
```

Fichier NUMERO

```
411
```

Fichier FICHRELANCE

```
.jg
.md140
Paris, le (DATE)                                    RAPPEL (NUMERO)


(APPEL) (PRENOM) (NOM), (ADR).


.MG10
.MD84
            (APPEL),


En date du (DATE), je vous ai adressé ma facture n°(NUMERO),
(CONCERNANT).
Or à ce jour,aucun réglement ne m'est parvenu.


Je pense qu'il  s'agit d'un oubli de votre part, et vous prie
de bien vouloir m'adresser la somme de :  (MONTANT) Fr  par
tout moyen à votre convenance.


Dans l'attente,
Je vous prie d'agréer, (APPEL), mes salutations distinguées.


                                                  Signature
.mp0
.mg0
.FF
```

Fichier FICHCOMMANDE

```
.mg2
.JG
.md90
Paris, le (DATE)                              COMMANDE N°(NUMERO)


.Md140
(NOM), (ADR).
.Md90

.MG10

$COMMANDES$

         Réf     Désignation                 Quantité     Unité


.MG4
.Identité bancaire .... :  code étab. guichet  compte    clé RIB
Banque  XX                 xxxxxx     xxxx    xxxxxxxxxxx  xxxx

Répertoire des métiers :  xxxxxxxxx
Sirène ............... :  xxxxxxxxxxxxx
.MG0
.FF
```

Graphisme sur Minitel

Patrice Neveu

Nouvelle étape des recherches de Pom's dans le domaine de la communication, voici un programme de conversion Image HGR/écran semi-graphique Minitel, ce qui — compte tenu des contraintes Vidéotex (2 couleurs seulement par groupe de 6 pavés) — représente une recherche remarquable.

Cette nouvelle étape, après le détecteur de sonnerie du numéro 32 représente un pas de plus vers une prochaine application conversationnelle entre votre ordinateur et le Minitel de vos correspondants : l'Apple // serveur...

S'abonner

Vous cherchiez une bonne raison de vous abonner ? en voici quelques-unes :

% six numéros pour le prix de cinq ;
♥ l'assurance de n'en rater aucun ;
✆ une assistance technique prioritaire, par téléphone, par Calva-Com ;
✉ Pom's dans votre boîte aux lettres avant mise en kiosque ;
➸ les petites annonces gratuites ;
✌ un tirage au sort à compter du 30 novembre pour recevoir logiciels de valeur, documentations techniques...

Tandis que nous bénéficions de superbes images en haute résolution sur nos Apple //, le Minitel connecté de fraîche date continue à faire triste mine avec son texte et ses médiocres images. Il suffit pourtant d'un programme de conversion pour profiter de ses 8 nuances de gris et de ses mosaïques semi-graphiques.

Quoi de mieux pour surprendre un utilisateur qu'une *Lady* numérisée, ou votre dernière œuvre graphique réalisée souris à la main ?

Méthode

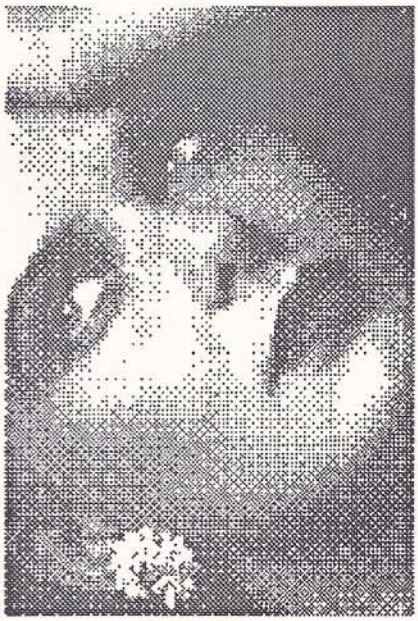
Mais ces images HGR ont une définition de 280 x 192 points, tandis que notre petit Minitel ne peut en afficher que 80 x 72. Il y aura certes perte importante de définition, mais vous serez surpris par ce dont ces 1440 points sont capables après être remplis par la routine HGR.MNTL.

Ce programme est conçu pour pouvoir :

- convertir les images HGR en mosaïques Minitel à plusieurs niveaux de gris, plus ou moins contrastés, ou en noir et blanc ;
- sauver le résultat sur disque ;
- le récupérer ultérieurement.

La conversion se déroule en deux phases :

La première est écrite en langage-machine et analyse l'image HGR pour donner le nombre de points par zones de 3 x 3 ou de 4 x 3 points, stocker ces chiffres, puis indiquer de combien de points est constituée l'image entière.



Chaque petite zone formera un carré sur l'écran du Minitel. Or, comme l'indique la page 10 du Pom's n° 29, chaque caractère Minitel contient 6 de ces carrés.

Ainsi, les 280 colonnes HGR deviendront 80 colonnes Minitel, d'alternativement 3 puis 4 points HGR de large. Ce compte curieux est nécessaire à cause du stockage même des points HGR en

mémoire : 7 points par octet, et 40 octets par ligne que l'on doit mettre en relation avec les 40 caractères par ligne du Minitel.

Quant aux lignes, nous n'utiliserons que 192 lignes HGR (3 lignes par carré mosaïque), soit 64 lignes mosaïque, ou encore 21 lignes de caractères Minitel.

Donc horizontalement,
- 1 ligne Minitel = 40 caractères Minitel
- 1 caractère Minitel = 2 colonnes mosaïques
- 1 colonne mosaïque = 3 ou 4 colonnes HGR

et verticalement,
- 1 colonne Minitel = 24 caractères Minitel
- 1 caractère Minitel = 3 lignes mosaïques
- 1 ligne mosaïque = 3 lignes HGR

La seconde phase est écrite en Applesoft car elle comporte de nombreux calculs. Il faut en effet, à partir de la densité par carré mosaïque. en déduire les deux couleurs que le caractère Minitel prendra.

Un caractère peut en effet prendre deux couleurs : une pour le caractère lui-même, l'autre pour le fond. Pour des mosaïques, cela signifie que certains des six carrés du caractère auront l'une ou l'autre couleur.

C'est là que le problème majeur intervient : comment calculer ces deux couleurs parmi les huit disponibles pour que l'image reste cohérente d'un caractère à l'autre ?
La solution passe par un calcul d'écart-type des six carrés d'un caractère :

$$ET = \sqrt{\frac{\sum x^2 - \frac{(\sum x)^2}{n}}{n - 1}}$$

On calcule aussi la densité moyenne MC de chaque carré de l'image par la division du nombre de points d'un carré sur leur nombre maximum (7 x 9).

Cela nous permet d'obtenir les deux couleurs du caractère:
la 1ère : MC – ET / 2
la 2ème : MC + ET / 2

Enfin, pour savoir si un carré prendra l'une ou l'autre couleur, on compare la densité moyenne du carré MC à la densité moyenne de l'image entière M.

La couleur est donnée par les séquences suivantes :

$1B $4x : couleur du caractère,
$1B $5x : couleur du fond.

x prend les valeurs
0 : noir
1 : rouge
2 : vert
3 : jaune
4 : bleu
5 : magenta
6 : cyan
7 : blanc

En tons de gris, nous avons du plus clair au plus foncé : 7 3 6 2 5 1 4 0.

Dernier raffinement, il est possible de changer le contraste normal de l'image (1), pour obtenir des effets spéciaux. Entrez donc 10 ou 30...

On remarque avec l'habitude que certaines images passent mal en niveaux de gris. On peut alors les passer en monochrome noir et blanc, ce qui les avantage généralement.

Comment faire ?

Trois éléments sont nécessaires :
un Minitel,
un câble de liaison à l'ordinateur,
un port série.

Le Minitel

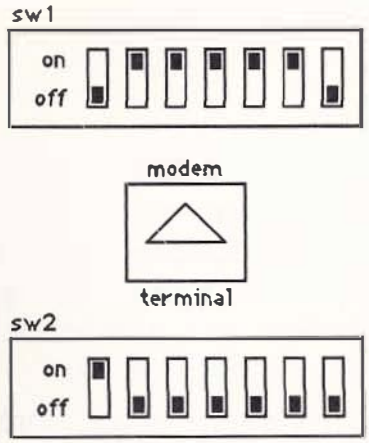
S'il est retournable (bi-standard) vos œuvres pourront être transmises à un correspondant équipé également d'un Minitel.

Le câble

La revue en tient à votre disposition si vous n'en disposez pas.

Le port série

Sur un //c ou un IIGS, il est intégré à la machine. Sur un][+ ou un //e, la carte SuperSérie Apple doit être insérée dans le slot 2 (si elle est dans un autre slot, modifiez la ligne 1100 du programme Basic), configurée de la façon suivante :



Sur votre IIGS, à l'aide du tableau de bord, mettre toutes les options par défaut du port modem, sauf :

```
Data/Stop Bits : 7/1
Parity : Even
DCD Handshake : No
DSR/DTR Handshake : No
```

Sur votre //c, supprimer le REM de la ligne 4.

Le programme

Pour utiliser ce programme de conversion il faut, si vous ne disposez pas de la disquette d'accompagnement, saisir et sauvegarder les programmes Basic HGR.MNTL et COMPACTEUR ainsi que la routine binaire DENSITE.

Après l'ordre :

```
RUN HGR.MNTL
```

le menu vous propose la conversion d'une image HGR qui consiste en sa transformation en un fichier TEXT à envoyer au Minitel. Dans cette option, après affichage du catalogue, indiquez le nom de votre image HGR (veillez à ce que son nom n'excède pas 8 caractères, éventuellement renommez-la avant).
Après transformation, un fichier TEXT a été créé sur disque :

```
nom_de_l'image.MNTL
```

Ce fichier peut déjà être transmis

au Minitel à l'aide de l'option *récupération.*

Le fichier qui est composé de 4206 caractères peut être réduit à l'aide du programme COMPACTEUR qui a pour objet la suppression de codes de changement de couleurs inutiles : à n'exécuter que lorsque l'image transformée a le contraste voulu.

Remarques

Si votre image HGR s'appelle :

LADY par exemple,

le fichier de texte créé par Hgr.Mntl s'appelle :

LADY.MNTL

et, après compactage par le programme *Compacteur* :

LADYC.MNTL

Pour envoyer votre belle image à votre Minitel depuis un programme Basic, s'inspirer des lignes 63002 à 63120 du programme *Hgr.Mntl.*

Pour l'envoyer à un correspondant qui possède un Minitel, s'inspirer du programme ENVOIIM. Le seul travail dudit correspondant sera de connecter son Minitel lorsqu'il entendra la porteuse.

Hgr.Mntl fonctionne aussi bien sous ProDOS que sous DOS 3.3.

La routine en Assembleur a été écrite avec Merlin-Pro.

Programme HGR.MNTL

Pour l'Apple //c, voir article.
Pour enfaciliter le comptage, certains espaces ont été remplacés par des 'puces' •

Pour l'Apple //c, supprimer le REM de la ligne 4.

```
0  REM
     Programme de transformation
     d'une image HGR en mosaique
     Minitel
4  REM PRINT  CHR$(4)"PR£2" : PRINT  CHR$
   (1)"1D"CHR$(1)"3P"CHR$(1)"8B
5  PRINT  CHR$ (4)"PR£3
6  PRINT
10  TEXT : HOME : INVERSE : PRINT "  Prog
   ramme de conversion d'une image HGR en
    Minitel  /  Patrice Neveu (87)      ":
    NORMAL
20  POKE 34,2
30  VTAB 5
40  POKE 768,104: POKE 769,168: POKE 770,
   104: POKE 771,166: POKE 772,223: POKE
   773,154: POKE 774,72: POKE 775,152: PO
   KE 776,72: POKE 777,96
1000  HIMEM: 38400
1005  ONERR  GOTO 63200
1010  PRINT  CHR$ (4)"BLOADDENSITE
1015  POKE 216,0
1020  LOMEM: 21760: REM   Pour protéger l
   a page 1 Haute Résolution
1030  HIMEM: 36864: REM   et la routine d
   es variables et fichiers.
1040  RESTORE : DIM ER$(17): FOR I = 1 TO
    8: READ R$: NEXT I
1050  FOR I = 1 TO 16: READ ER$:ER$(I) =
   ER$: NEXT I
1100 SL = 2
10000  REM
     Choix des différentes
    options de transfert
10010  VTAB 5
10020  PRINT "Voulez-vous (T)raiter une i
   mage HGR•••••••••••••••••••••••••••••
   •••••••••••••••••••••••••••(R)écupérer
    une image Minitel••••••••••••••••••
   •••••••••••••••••••••••••••••(Q)uitt
   er vers l'Applesoft
10025  PRINT "••••••••••••(S)ortir vers u
   n redémarrage ProDOS ";: GET R$
10030  PRINT R$: IF R$ = "R" THEN 63000
10033  IF R$ = "Q" THEN  TEXT : HOME : EN
   D
10036  IF R$ = "S" THEN  PRINT  CHR$ (4)"
   BYE"
10040  IF R$ <  > "T" THEN 10010
10043  ONERR  GOTO 63300
10045  PRINT  CHR$ (4)"CATALOG": PRINT
10050  INPUT "Quelle image voulez-vous tr
   aiter ? ";I$
10060  HGR : PRINT  CHR$ (4)"BLOAD"I$",A$
   2000
```

Programme ENVOIIM

```
100 ESC$ =  CHR$ (27):PRO1$ = ESC$ +  CHR$
     (57):DEC$ =  CHR$ (103):CNEX$ =  CHR$
     (104):RET1$ =  CHR$ (111):D$ =  CHR$
    (4)
110 IM$ = "LADYC.MNTL
120  PRINT D$"PR£2
125  PRINT  CHR$ (1)"Z
130  PRINT PRO1$RET1$PRO1$CNEX$
140  FOR I = 1 TO 6000: NEXT
150  PRINT D$"OPEN"IM$
160  PRINT D$"READ"IM$
170  ONERR  GOTO 200
180  GET R$: PRINT R$;
190  GOTO 180
200  PRINT : PRINT D$"CLOSE"IM$
210  PRINT PRO1$DEC$
220  PRINT D$"PR£3
```

Sur l'Apple //c, ajouter :

```
122  PRINT  CHR$(1)"1D"CHR$(1)"3P"CHR$(1)"
    8B
```

```
10070  TEXT
10080  PRINT "Voulez-vous de la couleur ?
    ";: GET R$: PRINT R$: IF R$ = "N" THE
   N MN% = 1: GOTO 40000
10090  INPUT "Quel contraste ? ";CO%
40000  REM 1ère phase en Langage-Machine
   et prépare la 2ème en Basic
40010  PRINT  CHR$ (4)"STORE DATA.MNTL
40020  CLEAR
40030  CALL 38144
40040  CLEAR
40050  PRINT  CHR$ (4)"RESTORE DATA.MNTL
40060 T =  PEEK (24) +  PEEK (25) * 256
40070  PRINT T / 5120:M% =  INT (T / 5120
   )
40080  PRINT  CHR$ (4)"PR£"SL
40090  PRINT  CHR$ (12): PRINT  CHR$ (14)
40100  DIM DT%(840,2)
50000  REM
```

Phase Applesoft de détermination de couleurs et points

```
50010  FOR Y = 0 TO 62 STEP 3: FOR X = 0
   TO 79 STEP 2
50020 AD% = 16384 + X
50030 C%(1) =  PEEK (AD% + Y * 80):C%(2)
   =  PEEK (AD% + 1 + Y * 80):C%(3) =  PE
   EK (AD% + (Y + 1) * 80):C%(4) =  PEEK
   (AD% + 1 + (Y + 1) * 80):C%(5) =  PEEK
    (AD% + (Y + 2) * 80):C%(6) =  PEEK (A
   D% + 1 + (Y + 2) * 80)
50040  IF MN% THEN CF% = 7:CL% = 0: GOTO
   50130
50050 S = C%(1) + C%(2) + C%(3) + C%(4) +
    C%(5) + C%(6)
50060 ET = ((C%(1) ^ 2 + C%(2) ^ 2 + C%(3
   ) ^ 2 + C%(4) ^ 2 + C%(5) ^ 2 + C%(6)
   ^ 2) - (S ^ 2) / 6) / 5
50070 ET = ET * CO%
50080 CL% = S / 6 - ET / 2:CF% = S / 6 +
   ET / 2
50090  IF CL% > 7 THEN CL% = 7: GOTO 5011
   0
50100  IF CL% < 0 THEN CL% = 0
50110  IF CF% > 7 THEN CF% = 7: GOTO 5013
   0
50120  IF CF% < 0 THEN CF% = 0
50130 C% = 32 + (C%(1) > M%) + 2 * (C%(2)
    > M%) + 4 * (C%(3) > M%) + 8 * (C%(4)
    > M%) + 16 * (C%(5) > M%) + 64 * (C%(
   6) > M%)
50140  RESTORE : FOR I = 0 TO CF%: READ F
   %: NEXT I:CF% = F%
50150  RESTORE : FOR I = 0 TO CL%: READ L
   %: NEXT I:CL% = L%
50160  PRINT  CHR$ (27) +  CHR$ (64 + CF%
   ) +  CHR$ (27) +  CHR$ (80 + CL%); CHR
   $ (C%);
50170 DT%(O%,0) = CF%:DT%(O%,1) = CL%:DT%
   (O%,2) = C%:O% = O% + 1
50180  NEXT X,Y
50190  PRINT  CHR$ (15)
50200  PRINT  CHR$ (4)"PR£3
50210  DATA  0,4,1,5,2,6,3,7
62000  REM
```

Sauvergarde de la mosaique sur Disque

```
62010  NORMAL : TEXT : HOME
62020  PRINT "Voulez-vous sauver cette im
   age Minitel sur disque ? ";: GET R$
62030  IF R$ = "N" THEN  RUN
62040  IF R$ <  > "O" THEN 62010
62050  PRINT  CHR$ (4)"OPEN"I$".MNTL
62060  PRINT  CHR$ (4)"WRITE"I$".MNTL
62070  PRINT  CHR$ (12): PRINT  CHR$ (14)
62080  FOR I = 0 TO O% - 1
62090  PRINT  CHR$ (27) +  CHR$ (64 + DT%
   (I,0)) +  CHR$ (27) +  CHR$ (80 + DT%(
   I,1)) +  CHR$ (DT%(I,2));
62100  NEXT I
62110  PRINT  CHR$ (15)
62120  PRINT  CHR$ (4)"CLOSE
62130  RUN
63000  REM
```

Affichage d'une mosaique sauvée sur Disque

```
63002  ONERR  GOTO 63300
63004  PRINT  CHR$ (4)"CATALOG
63010  INPUT "Quelle image voulez-vous ré
   cupérer ? ";I$
63020  PRINT  CHR$ (4)"UNLOCK"I$".MNTL
63040  PRINT  CHR$ (4)"PR£"SL
63050  PRINT  CHR$ (4)"OPEN"I$".MNTL
63060  PRINT  CHR$ (4)"READ"I$".MNTL
63070  ONERR  GOTO 63090
63080  GET R$: PRINT R$;: GOTO 63080
63090  PRINT  CHR$ (4)"CLOSE
63100  NORMAL
63110  PRINT  CHR$ (4)"PR£3
63120  POKE 216,0
63130  RUN
63200  REM
```

Erreur, il n'y a plus le disque d'origine

```
63210  HOME : PRINT : PRINT "Vous devriez
    remettre le disque de Conversion": PR
   INT
63220  GET R$
63230  POKE 216,0
63240  RUN
63300  REM
```

Erreur de disque, on recommence l'opération

```
63302  POKE 216,0
63303  TEXT
63305  CALL 768
63310 ER =  PEEK (222): HOME : TEXT
63320 RL =  PEEK (218) +  PEEK (219) * 25
   6
```

DOS 3.3 ProDOS

```
63360  PRINT "("ER") "ER$(ER)" ligne "RL
63370  GET R$
63390  RUN
63399  REM
```

Données contenant les messages d'erreur du DOS plus un pour l'Applesoft.

```
63400  DATA  "Langage non disponible","Paramètre hors limite","Rien n'est connecté" ,"Disquette protégée a l'écriture","Fin des données","Fichier introuvable","Erreur de paramètre","Erreur d'Entrée - Sortie","Disque plein"
63405  DATA  "Fichier verrouillé"
63410  DATA  "Pas de mémoire tampon disponible","Erreur quant au type de fichier","Programme trop important","Cette commande ne peut etre passée en mode direct"
63420  DATA  "Erreur de syntaxe","Erreur dans l'Applesoft"
```

Programme COMPACTEUR

```
5  HOME
10 D$ =  CHR$ (4): DIM CAR(4205),ST(4),C(4):ES$ =  CHR$ (27):ES = 27
20  INPUT "Fichier à compacter ? ";F$:F1$ = F$ + "C" + ".MNTL":F$ = F$ + ".MNTL
29  PRINT "Chargement du fichier "F$
30  PRINT D$"OPEN"F$: PRINT D$"READ"F$: FOR I = 0 TO 4205: GET CAR$:CAR(I) =  ASC (CAR$): NEXT : PRINT D$"CLOSE
80  PRINT "Compactage en cours
90  PRINT D$"OPEN"F1$: PRINT D$"WRITE"F1$
100 PTR =  - 1
110 PTR = PTR + 1: IF PTR > 4205 THEN 400
120 C(1) = CAR(PTR): IF C(1) <  > ES THEN  PRINT  CHR$ (C(1));: GOTO 110
130 PTR = PTR + 1
140 C(2) = CAR(PTR): IF C(2) < 64 OR C(2) > 71 THEN  PRINT ES$ CHR$ (C(2));: GOTO 110
150 PTR = PTR + 1
160 C(3) = CAR(PTR): IF C(3) <  > ES THEN  PRINT ES$ CHR$ (C(2)) CHR$ (C(3));: GOTO 110
170 PTR = PTR + 1
180 C(4) = CAR(PTR): IF C(4) < 80 OR C(4) > 87 THEN  PRINT ES$ CHR$ (C(2))ES$ CHR$ (C(4));: GOTO 110
190 OK = 1: FOR I = 1 TO 4: IF ST(I) <  > C(I) THEN OK = 0
200  NEXT : IF OK THEN  GOSUB 1000: GOTO 110
210  FOR I = 1 TO 4: PRINT  CHR$ (C(I));: NEXT : GOSUB 1000: GOTO 110
400  PRINT D$"CLOSE": PRINT "Fichier "F1$" créé": PRINT "Terminé.": END
1000  FOR I = 1 TO 4:ST(I) = C(I): NEXT : RETURN
```

Raccourcir WPL

Pour comprimer un programme WPL trop long (la limite se situe à 2048 caractères), toutes les commandes qui s'exécutent sans "return" peuvent être accollées.
Le programme STARTUP.N de la page 10 pourrait ainsi s'écrire comme ci-dessous et passer de 288 caractères à 244.

```
<esc esc>PND
PPRun instant !
SVIDE
O
OA£
?BF* —:—      O*
?DWNO
DXDF* *=$D, NO, LNUMERO, HH, F<??<$C<, O?, S=, PDOINIDATE.N*
O?F!/!£ PAS19!
O?F!/!£ PAS!
O?BF! ! PAS!
O?F*, *=$C, *
O?F*, *=$B, *
O?F<$<$<A
SVARIDATE.N
O
PDOVARIDATE.N
```

Récapitulation DENSITE

Après avoir saisi cette récapitulation sous moniteur, vous la sauvegarderez par : BSAVE DENSITE, A$9500, L181

```
9500:A9 40 8D 5A 95 A9 00 85
9508:07 85 06 8D 59 95 85 18
9510:85 19 A9 00 85 09 A9 03
9518:8D B4 95 A5 07 A2 00 A0
9520:00 20 11 F4 A5 06 4A A8
9528:B1 26 90 08 4A 4A 4A 29
9530:0F 4C 36 95 29 07 A8 B9
9538:A4 95 18 65 09 85 09 E6
9540:07 CE B4 95 D0 D5 A5 06
9548:4A B0 08 A4 09 B9 8D 95
9550:4C 58 95 A4 09 B9 97 95
9558:8D 00 40 EE 59 95 D0 03
9560:EE 5A 95 A5 09 18 65 18
9568:85 18 90 02 E6 19 A4 07
9570:88 88 88 84 07 E6 06 A5
9578:06 C9 50 90 95 A9 00 85
9580:06 A4 07 C8 C8 C8 84 07
9588:C0 BF 90 86 60 00 01 01
9590:02 03 04 05 06 06 07 00
9598:01 01 02 02 03 04 04 05
95A0:05 06 06 07 00 01 01 02
95A8:01 02 02 03 01 02 02 03
95B0:02 03 03 04 00
```

Source DENSITE.S

Assembleur Merlin-Pro

][+
//e
//e+
//c
][gs

```
        LST     OFF
        ORG     $9500

*****************************************
* Objet général:                        *
*                                       *
* Stockage de la concentration des points *
* d'un sixième de caractère en mémoire. *
*                                       *
*****************************************

*****************************************
* Lot de points fait 3x3 points (pair)  *
*                    3x4        (impair) *
*                                       *
* Donc 21 lignes Minitel seront utilisées *
*****************************************

* Labels

XCOORD  =       $06         ; Coordonnées du lot à étudier, 0-79
YCOORD  =       $07         ; Coordonnées du lot à étudier, 0-191
NBRPTS  =       $09         ; Nombre de points dans le lot
PTSTOT  =       $18
HBAS    =       $26         ; Adresse sortie de HPOSN
HIMEM   =       $73         ; Protection du programme face aux variables

HPOSN   =       $F411       ; Trouve l'adresse d'un groupe de points

* Boucle des lignes et des colonnes HGR
*--------------------------------------

        LDA     £$40        ; Les densités de points seront mises
        STA     STOCKDENS+2; à partir de $4000
        LDA     £0
        STA     YCOORD      ; On commence en haut à gauche de l'image
        STA     XCOORD
        STA     STOCKDENS+1
        STA     PTSTOT      ; Mise à zéro du nombre de points constituant
        STA     PTSTOT+1    ; l'image et utilisé dans la moyenne générale.

* Programme de calcul de concentration d'un lot de points
*-------------------------------------------------------

CALCLOTS
        LDA     £0          ; Mise à zéro du nombre de points d'un lot.
        STA     NBRPTS
        LDA     £3
        STA     NBRLGNS     ; Nombre de lignes dans le lot

LITOCTET
        LDA     YCOORD      ; On va se positionner au début de la ligne
        LDX     £0          ; dont le numéro est dans YCOORD.
        LDY     £0
        JSR     HPOSN       ; Va chercher son adresse

        LDA     XCOORD
        LSR                 ; Colonne / 2 = numéro d'octet
        TAY
        LDA     (HBAS),Y    ; Prend l'octet sur la bonne ligne
        BCC     PTSG        ; Pour savoir si on est dans la colonne de
        LSR                 ; gauche ou de droite d'un caractère Minitel.
        LSR                 ; Dans le cas de la partie gauche, décale à
        LSR                 ; droite les 4 bits intéressants.
        AND     £%00001111  ; Et on les isole
        JMP     ADDPTS
PTSG    AND     £%00000111  ; On isole les 3 points intéressants de droite
ADDPTS
        TAY                 ; Transforme la valeur en index
        LDA     NBRPRVAL,Y  ; Prend le nombre de bits selon la valeur
        CLC
        ADC     NBRPTS      ; Additionne ça au nombre de points du lot
        STA     NBRPTS

        INC     YCOORD      ; Passe à la prochaine ligne HGR
        DEC     NBRLGNS     ; Ça fait une ligne de moins à faire
        BNE     LITOCTET    ; sur les 3 que comple 1/6 de caractère MINTL

* Il y a huit concentrations possibles

        LDA     XCOORD      ; À nouveau il faut faire la distinction
        LSR                 ; entre la partie gauche et la partie droite
        BCS     PTS12       ; de l'octet considéré.

        LDY     NBRPTS      ; Dans un cas on prend une table contenant
        LDA     COULEUR,Y   ; 10 concentrations possibles (0 à 9)
        JMP     STOCKDENS

PTS12   LDY     NBRPTS      ; Dans l'autre cas on a 13 possibilités
        LDA     COULEUR2,Y  ; (0 à 12)

STOCKDENS STA   $4000       ; On sauve le nombre de points dans une table
        INC     STOCKDENS+1; Puis on passe à la prochaine position...
        BNE     STOCKED
        INC     STOCKDENS+2
STOCKED
        LDA     NBRPTS      ; Pour avoir le nombre total de points
        CLC                 ; constituant l'image, on rajoute ceux-là...
        ADC     PTSTOT
        STA     PTSTOT
        BCC     NOC
        INC     PTSTOT+1
NOC
        LDY     YCOORD      ; On fait cela car à chaque 1/6 ème de
        DEY                 ; caractère, le n° de ligne augmente de 3
        DEY                 ; Or, on ne doit passer aux 3 lignes HGR
        DEY                 ; suivantes qu'en bout de ligne, c'est à
        STY     YCOORD      ; dire...

        INC     XCOORD
        LDA     XCOORD
        CMP     £80         ; Lorsque XCOORD est à 80. (40x2)
        BCC     CALCLOTS

        LDA     £0          ; Alors seulement on peut sauter les 3 lignes
        STA     XCOORD      ; que l'on vient de traiter et revenir en
        LDY     YCOORD      ; début de ligne.
        INY
        INY
        INY
        STY     YCOORD
        CPY     £191        ; La ligne HGR n°191 est la dernière
        BCC     CALCLOTS
        RTS                 ; Après laquelle on passe à la phase n°2

* Tables diverses

* On peut changer ces deux premières tables qui indiquent
* que pour un nombre X de points dans le 1/6 de caractère
* on a le niveau de gris Y.

COULEUR  HEX    00010102030405060607
COULEUR2 HEX    000101020203040405060607

* Cette table indique que pour la valeur X, Y bits sont à 1

NBRPRVAL HEX    00010102010202030102020302030304

NBRLGNS  HEX    00
```

GS & QuickDraw II

LE Pham Hiep

Voici un petit programme qui vous permet d'utiliser les fonctions graphiques Super haute résolution de l'Apple IIGS. Il vous montrera les principes de base pour exploiter les possibilités des "outils" à partir du Basic Applesoft.

Les possibilités graphiques du IIGS

L'Apple IIGS, grâce à sa ROM de 128Ko, présente de nombreuses possibilités qui ne sont pas exploitables directement depuis l'Applesoft. Dans cet article, nous nous intéressons surtout aux outils graphiques.

L'Apple IIGS est doté, en plus des traditionnels modes graphiques de l'Apple //, du mode Super-Hi-Res. Ce dernier est géré par *QuickDraw II*, il vous donne la possibilité de contrôler les résolutions suivantes :

- 320x200 ou
- 640x200 en 16 ou 4 couleurs (en standard) parmi une palette de 4096.

QuickDraw II sait gérer les fonctions de base telles :

- les lignes,
- les rectangles,
- le texte,

mais également les manipulations de l'environnement graphique comme :

- la gestion des coordonnées,
- la gestion des régions visibles...

Possibilités de mise en œuvre

Pour utiliser les outils du GS, vous disposez de plusieurs moyens :

- un langage évolué comme Pascal ou C,
- l'assembleur sous moniteur (qui fonctionne fort bien) ou APW (Apple Programmer Workshop).

Ici, nous allons utiliser l'assembleur pour définir une interface permettant au Basic d'accéder aux fonctions graphiques. Il faut savoir également que les outils ne sont pas indépendants. Dans le cas qui nous intéresse, pour faire fonctionner Quickdraw II, il faut également installer Locator, Memory (qui sont nécessaires pour le fonctionnement du système). Mais heureusement le Basic.system s'installe déjà avec ces derniers...

Le fonctionnement de certains des outils du système nécessite parfois quelques espaces mémoires réservées, dans le cas de QuickDraw II v1.1, il s'agit de 3 pages de 256 octets dans le premier *bank* de mémoire (bank 00).
Les appels aux fonctions des outils doivent se faire en mode natif pur (c'est-à-dire les drapeaux du registre d'état du 65816 sont : e=0, m=0, x=0).

Maintenant nous pouvons commencer à écrire notre petit programme qui permet :

1) l'installation de QuickDraw II ;
2) sa désactivation, nécessaire pour libérer les mémoires et assurer la cohérence du système ;
3) l'appel de certaines fonctions graphiques.

Appels des fonctions des outils

Les appels des fonctions se font à partir du point d'entrée $E10000 (adresse 0 du banc de mémoire $E1) de la façon suivante :

- mettre dans X le numéro de la fonction souhaitée puis,
- faire JSL $E10000.

Souvent, les paramètres sont passés par la pile ; par exemple, pour positionner le curseur, nous empilons sa position horizontale, sa position verticale, puis nous faisons :

```
LDX £$...
JSL $E10000
```

En assembleur (APW) il existe une bibliothèque de macros qui permet de faire des appels de fonctions des outils par leur nom :

```
_MoveTo, MMStartup, ...
```

Description du programme

Vous trouverez ici le programme source écrit sous APW avec les directives nécessaires pour son assemblage.

Quelques explications concernant deux macros usuelles :

pushword... empile un mot (c'est-à-dire 16bits). Son équivalent : *LDA... PHA*

pushlong... empile un mot long (32bits). Il est équivalent à deux push word

Toutes les macros commençant par le souligné "_" sont des appels à des fonctions des outils, en général, leur équivalent est :

```
LDX £$... ; n° de fonction
JSL $E10000
```

Le programme binaire

Le programme en assembleur se logera en $800. Les programmes Basic de ce fait commenceront en $1000.

De $800 à $AFF : espace réservé pour initialiser Quickdraw.

À partir de $B00 : les points d'entrées des fonctions graphiques ; ils vous facilitent les appels. À la suite de ces points d'entrées nous laisserons la place pour passer des paramètres.

Ici, nous avons adopté le passage des paramètres par des Pokes et les appels par des Calls pour 2 raisons :

- pour ceux qui voulaient utiliser QD.BIN en assembleur, la manière de les appeler reste identique ;
- il se peut que vous ne vouliez pas changer toutes les valeurs d'une fonction (exemple : vous voulez faire un trait horizontal, il suffit de ne changer que le paramètre H).

Le programme de démonstration QD.SAMPLE devrait éclairer les choses.

Concernant les listings

QD.SAMPLE : programme en Applesoft qui donne un exemple d'utilisation.

Pour ceux qui possèdent APW il faut taper :

- MAIN : programme (en EXE)
- QD.SRC : source en assembleur APW (en ASM65816)

puis lancer MAIN ; vous obtiendrez QD.BIN l'interface binaire nécessaire.

Pour ceux qui ne possèdent pas de APW vous devez tapez directement la récapitulation.

Comment utiliser QD.BASIC ?

Vous êtes arrivés au Basic.System en ayant booté sur ProDOS 16 puis MouseDesk ou Sélecteur, et sur votre disque vous avez QD.BIN. Pour disposer des fonctions graphiques, vous devrez vous inspirer du programme de démonstration QD.SAMPLE.

Pour charger QD.BIN et lui réserver la place, les lignes 9 à 20 relogent le Basic :

ligne 9 : est-on relogé ?
ligne 10 : fixe la nouvelle adresse de chargement du programme Basic.
ligne 15 : charge QD.BIN.
ligne 20 : recharge le programme Basic, mais à la bonne adresse.

Il vous faut alors définir une variable, QD par exemple, valant 2816 et les fonctions vous sont alors accessibles ainsi :

Initialisation

- CALL QD

Désactivation

- CALL QD+3

Positionnement

AT(H,V)

- Poker en QD+70 et QD+80 la position verticale (de 0 à 200)
- Poker en Q+68 et QD+69 la position horizontale (de 0 à 320), poids faible, poids fort
- CALL QD+6

Impression de chaînes

PRINT (T$)

- Poker en QD+72 la longueur de la chaîne
- Poker la chaîne à partir de QD+73 (maximum 127 caractères)
- CALL QD+9

Mode d'impression graphique

SetPenMode (C)

C vaut de 0 à 3 ou de $8000 à $8003.

- Poker C en QD+50 et QD+51 (faible, fort)
- CALL QD+12

Les valeurs de C sont les suivantes :

$0 COPY : la source écrase la destination
$1 OR : la source s'ajoute à la destination
$2 XOR : le Ou Exclusif entre la source et la destination sera affiché
$3 BIC : on affiche : (NOT source) AND destination
$4 mêmes modes que $0 à $3
$5 mais s'appliquent à l'affichage
$6 du texte : seul le corps du
$7 texte est concerné

À toutes ces valeurs, si on ajoute $8000 à C, la source est inversée avant opération logique : si C = $8001, (NOT source) OR destination sera affiché.

Couleur du crayon

SetSolidPenPat (C)

- Poker 0 en QD+51
- Poker C en QD+50
- CALL QD+18

Attributs du texte

SetTextFace (C)

C=0 normal
C=1 gras
C=2 souligné

- Poker 0 en QD+53
- Poker C en QD+52
- CALL QD+21

Mode d'impression texte

SetTextMode (C)

Idem SetPenMode pour le texte.

- Poker C en QD+54/QD+55
- CALL QD+24

Couleur du texte

SetForeColor (C)

- Poker 0 en QD+51
- Poker C (de 0 à 15) en QD+50
- CALL QD+27

Tracé d'une ligne

LineTo (H,V)

- Poker coordonnées d'arrivée V en QD+70 et QD+71
- Poker coordonnées d'arrivée H en QD+68 et QD+69
- CALL QD+30

Inversion d'un rectangle
InvertRect (V1, H1, V2, H2)

Le rectangle est défini par les coordonnées du point haut/gauche et bas/droit.

- Poker V1 en QD+56 et QD+57
- Poker H1 en QD+58 et QD+59
- Poker V2 en QD+60 et QD+61
- Poker H2 en QD+62 et QD+63
- CALL QD+33

Dessin d'un rectangle
PaintRect (V1, H1, V2, H2)

- Poker V1 en QD+56 et QD+57
- Poker H1 en QD+58 et QD+59
- Poker V2 en QD+60 et QD+61
- Poker H2 en QD+62 et QD+63
- CALL QD+36

Tracé d'un arc
PaintArc (V1, H1, V2, H2, S, C)

Le rectangle V1/H1/V2/H2 sert à définir l'ellipse support de l'arc, S est l'angle de départ, C l'angle de tracé.

- Poker V1 en QD+56 et QD+57
- Poker H1 en QD+58 et QD+59
- Poker V2 en QD+60 et QD+61
- Poker H2 en QD+62 et QD+63
- Poker S en QD+64 et QD+65
- Poker C en QD+66 et QD+67
- CALL QD+39

Couleur de fond
SetBackColor (C)

- Poker 0 en QD+51
- Poker C (de 0 à 15) en QD+50
- CALL QD+42

Bibliographie nécessaire

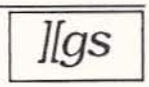
Clefs pour Apple IIGS (PSI)
Rom tools (Apple)

Programme QD.SRCE
Apw

```
*************************************
*   QD_Basic
*       par Le Pham  Hiep 87
*
*************************************
*  3 pages de quickdraw $800.$afff
*  programme en $b00
*************************************

        list off
        absaddr on
        GEN   off
        SYMBOL off
        org   $000b00

        KEEP QD

        mcopy qd.macros


QDstart     START
            jmp  initqd          ;installe Quickdraw
            jmp  qdoff           ; desactive Quickdraw (libère le système)
            jmp At               ;AT (H,V)
            jmp Print            ;Print (strptr) : de type Pascal
            jmp SetPenMode       ;SetPenMode(color) :0..3 ou $8000.$8003
            jmp GrafOff          ;GraffOff
            jmp SetSolidPenPat   ;SetSolidPenPat(color) :couleur crayon
            jmp SetTextFace      ;SetTextFace(TextFace) : 0..3,souligne
            jmp SetTextMode      ;SetTextMode(TextMode) : 0..3 ou b15=1
            jmp SetForeColor     ;SetForeColor(color) = set texte color
            jmp LineTo           ;LineTo(H,V)
            jmp InvertRect       ;InvertRect(rectptr)
            jmp PaintRect        ;PaintRect(rectptr)
            jmp PaintArc         ;PaintArc(rectptr,startangle,angle)
            jmp  SetBackcolor    ;SetBackColor(color) : couleur du fond
            jmp  nothing

MyId        dc i'00'             ;normalement $3001

color       dc i'05'             ;00..0f

TextFace    dc i'01'             ;00..03
TextMode    dc i'00'             ;00..03  $8000..$8003

rectptr     dc i'10,10,100,100';V1,H1,V2,H2
StartAngle  dc i'00'
ArcAngle    dc i'10'

H           dc i'20'
V           dc i'20'


strptr      str 'hello 0123'     ;chaine dutype Pascal(ascii bas)
            ds 120

buffer      dc h'EAEAEAEA'

*==============================*
```

FichierEXE
QD.MAIN

```
macgen qd.src qd.mac
    ros .d2/macros/=↵
asml qd.src↵
rename qd qd.bin↵
makebin qd.bin↵
```



```
nothing     anop                ;ne rien faire
            rts

*==============================*
emul        anop                ;revient en mode émulation
            sep £$30
            sec
            xce
            rts                 ;A X 8 bits

*==============================*
natif       anop                ;passage en mode natif
            clc                 ;
            xce                 ;natif
            rep £$30            ;A X 16
            rts

*==============================*
getmyID     anop                ;récupère le numéro
            phd                 ;IDentification affectée par PRODOS16

            pushlong £0
            pushlong £$800
            _Findhandle
            tsc
            tcd
            ldy £$6
            lda [$01],y
            sta >Myid
            pla
            pla

            pld

            rts

*==================================*
*** LES FONCTIONS                 ***
*==================================
initqd      anop
            jsr natif

            pushword £0
            _QDStatus
            pla
            bne deja_inst

            jsr getmyid

ToolsLoaded anop
            pushlong £0000
            pushlong £$e12000
            _FindHandle
            _DisposeHandle


cont1       anop
            PushWord £$800
            PushWord £00        ;320
            PushWord £160       ;nombre d'octets par ligne
            PushWord >MyID
            _QDStartup
            bcc qdstartok
            brk $ff
```

```
qdstartok   anop
            jmp emul            ;FIN QDinit


deja_inst   anop
            _GrafOn
            jmp emul
*==================================
qdoff       anop
            jsr natif
            _Grafoff
            _QDshutdown
            jmp emul            ;FINqdoff

*==================================
at          anop
            jsr natif

            pushword H
            pushword V
            _MoveTo

            jmp emul            ;
*==================================
print       anop
            jsr natif

            pushlong £strptr
            _DrawString

            jmp emul            ;fin Print
*==================================
SetPenMode  anop
            jsr natif

            pushword color
            _SetPenMode

            jmp emul
*==================================
GrafOff     anop
            jsr natif

            _GrafOff

            jmp emul
*==================================
SetSolidPenPat anop
            jsr natif

            pushword color
            _SetSolidPenPat

            jmp emul
*==================================
SetTextFace anop
            jsr natif

            pushword Textface
            _SetTextFace

            jmp emul
*==================================
SetTextMode anop
            jsr natif
```

```
            pushword textmode
            _SetTextMode

            jmp emul
*=====================================
SetForeColor anop
            jsr natif

            pushword color
            _SetForeColor
            jmp emul
*=====================================
LineTo      anop
            jsr natif

            pushword H
            pushword V
            _LineTo

            jmp emul
*=====================================
InvertRect  anop
            jsr natif

            pushlong £rectptr
            _InvertRect

            jmp emul            ;FIN drawrect
*=====================================
```

ProDOS 16

```
PaintRect   anop
            jsr natif

            pushlong £rectptr
            _PaintRect

            jmp emul
*=====================================
PaintArc    anop
            jsr natif

            pushlong £rectptr
            pushword StartAngle
            pushword ArcAngle
            _PaintArc

            jmp emul
*=====================================
SetBackColor anop
            jsr natif

            pushword color
            _SetBackColor

            jmp emul
*=====================================

            END
```

Programme QD.SAMPLE

```
1 ::::: REM ============================
2 ::::: REM ==  exemple d'utilisation ==
3 ::::: REM ==  de QuickDrawII du GS  ==
4 ::::: REM == sous Basic par LPH(87) ==
                (avec JV et DDP)
5 ::::: REM ============================
6 REM ** Variables utilisees par les SP :
7 REM    **   V,H,V1,H1,V2,H2,T$,S,C,QD
8 :
9  IF  PEEK (104) = 16 THEN 50
10  POKE 103,1: POKE 104,16: POKE 16 * 25
   6,0
15  PRINT  CHR$ (4);"bload qd.bin":
20  PRINT  CHR$ (4);"-qd.sample"
50  DIM TXT$(4):QD = 11 * 256: PRINT  CHR
   $ (4);"pr£3": PRINT : PRINT : PRINT "
          Exemples d'utilisation de Quic
   kdrawII sous le Basic"
51 TXT$(1) = "Copy":TXT$(2) = "Or":TXT$(3
   ) = "Xor":TXT$(4) = "Bic"
60 : GOTO 10000
100 :
1000 : CALL QD: REM InitQuickdraw

1010  RETURN
1015 :
1100 : CALL (QD + 3): REM QuickdrawOff
1110  RETURN
1120 :
1200 : REM $$$$ At (H ,V )
        H : de 00 à 320
        V : de 00 à 200
1210  POKE (QD + 71),V / 256: POKE (QD +
   70),(V - 256 *  PEEK (QD + 71))
1220  POKE (QD + 69),H / 256: POKE (QD +
   68),(H - 256 *  PEEK (QD + 69))
1230  CALL (QD + 6):
1240  RETURN
1299 :
1300 : REM $$$$ Print (t$)
1310  POKE (QD + 72), LEN (T$):
1320  FOR I = 1 TO  LEN (T$)
1330  POKE (QD + 72 + I), ASC ( MID$ (T$,
   I,1)):
1340  NEXT : CALL (QD + 9)
1350  RETURN
1399 :
1400 : REM $$$$ SetPenMode (C)
          00..03  $8000..$8003
1410  POKE (QD + 51),0
1420  IF C > 128 THEN  POKE (QD + 51),128
   :C = C - 8 * 16 * 256:
1430  POKE (QD + 50),C
1440  CALL (QD + 12)
1450  RETURN
1499 :
1600 : REM $$$$ SetSolidPenPat (C)
                        00..15
1610  POKE (QD + 51),0: POKE (QD + 50),C:
    CALL (QD + 18)
1620  RETURN
1699 :
```

```
1700 : REM $$$$ SetTextFace (C)
  00..03 suivant la version de QDII
1710  POKE (QD + 53),C / 256: POKE (QD +
   52),(C - 256 *  PEEK (QD + 53))
1720  CALL (QD + 21)
1730  RETURN
1799 :
1800 : REM $$$$ SetTextMode(C)
          0..3  $8000..$8003
1810  POKE (QD + 55),C / 256: POKE (QD +
   54),(C - 256 *  PEEK (QD + 55))
1820  CALL (QD + 24)
1830  RETURN
1899 :
1900 : REM $$$$ SetForeColor(C)
                    00..15
1910 : POKE (QD + 51),0: POKE (QD + 50),C
   : CALL (QD + 27)
1930  RETURN
1999 :
2000 : REM $$$$ LineTo(H,V)
2010  POKE (QD + 71),V / 256: POKE (QD +
   70),(V - 256 *  PEEK (QD + 71))
2020  POKE (QD + 69),H / 256: POKE (QD +
   68),(H - 256 *  PEEK (QD + 69))
2030  CALL (QD + 30)
2040  RETURN
2099 :
2100 REM $$$$ InvertRect(V1,H1,V2,H2)
2110  POKE (QD + 57),V1 / 256: POKE (QD +
    56),(V1 - 256 *  PEEK (QD + 57))
2120  POKE (QD + 59),H1 / 256: POKE (QD +
    58),(H1 - 256 *  PEEK (QD + 59))
2130  POKE (QD + 61),V2 / 256: POKE (QD +
    60),(V2 - 256 *  PEEK (QD + 61))
2140  POKE (QD + 63),H2 / 256: POKE (QD +
    62),(H2 - 256 *  PEEK (QD + 63))
2150  CALL (QD + 33)
2160  RETURN
2199 :
2200 REM $$$$ PaintRect(V1,H1,V2,H2)
2210  POKE (QD + 57),(V1 / 256): POKE (QD
    + 56),(V1 - 256 *  PEEK (QD + 57))
2220  POKE (QD + 59),H1 / 256: POKE (QD +
    58),(H1 - 256 *  PEEK (QD + 59))
2230  POKE (QD + 61),V2 / 256: POKE (QD +
    60),(V2 - 256 *  PEEK (QD + 61))
2240  POKE (QD + 63),H2 / 256: POKE (QD +
    62),(H2 - 256 *  PEEK (QD + 63))
2250  CALL (QD + 36)
2260  RETURN
2299 :
2300 REM $$$$ PaintArc(V1,H1,V2,H2,S,
   C): rectangle,startangle,angle
2301 :
2310  POKE (QD + 57),(V1 / 256): POKE (QD
    + 56),(V1 - 256 *  PEEK (QD + 57))
2320  POKE (QD + 59),H1 / 256: POKE (QD +
    58),(H1 - 256 *  PEEK (QD + 59))
2330  POKE (QD + 61),V2 / 256: POKE (QD +
    60),(V2 - 256 *  PEEK (QD + 61))
2340  POKE (QD + 63),H2 / 256: POKE (QD +
    62),(H2 - 256 *  PEEK (QD + 63))
2350  POKE (QD + 65),S / 256: POKE (QD +
   64),S - 256 *  PEEK (QD + 65)
2360  POKE (QD + 67),C / 256: POKE (QD +
   66),C - 256 *  PEEK (QD + 67)
2370  CALL (QD + 39)
2390  RETURN
2399 :
2400 : REM $$$$ SetBackColor(C)
                 00..15
2410  POKE (QD + 51),0: POKE (QD + 50),C:
2420  CALL (QD + 42)
2430  RETURN
2499 :
5000 REM ++++ Cartouche(V1,H1,V2,H2,C
   )  bouton type ombre
5005 S = C:C = 0: GOSUB 1600
5010 V1 = V1 + 2:H1 = H1 + 2:V2 = V2 + 2:
   H2 = H2 + 3: GOSUB 2200
5015 C = S: GOSUB 1600
5020 V1 = V1 - 2:H1 = H1 - 2:V2 = V2 - 2:
   H2 = H2 - 3: GOSUB 2200
5025 C = 15 - S: GOSUB 1600
5030 V1 = V1 + 1:H1 = H1 + 1:V2 = V2 - 1:
   H2 = H2 - 1: GOSUB 2200
5035 V1 = V1 - 1:H1 = H1 - 1:V2 = V2 + 1:
   H2 = H2 + 1
5040 C = S: GOSUB 1600
5090  RETURN
5099 :
5100 : REM ++++ Centertexte(t$)
5150 H = H1 + (H2 - H1) / 2 - ( LEN (T$)
   * 3.5)
5160 V = V1 + (V2 - V1) / 2 + 3
5170  GOSUB 1200:
5180  GOSUB 1300
5190 : RETURN
5199 :
9000 :
9010 :
10000  GOSUB 1100: GOSUB 1000: REM
******** DEBUT DE L'EXEMPLE ********
10010 C = 5: GOSUB 1600
10020 V1 = 0:H1 = 0:V2 = 200:H2 = 320: GO
   SUB 2200
10030 :
10040  FOR V1 = 0 TO 198
10050 C = C + 1: GOSUB 1600: IF C = 15 TH
   EN C =  - 1
10060 V2 = V1 + 2: GOSUB 2200
10070  NEXT
10080 :
10090  FOR C = 0 TO 3: GOSUB 1400: REM
   chgt PenMode
10095 V1 = C * 50:V2 = V1 + 50:H2 = 320
10100  FOR H1 = 319 TO 0 STEP  - 1
10110 C = C + 1: GOSUB 1600: IF C = 15 TH
   EN C =  - 1
10120  GOSUB 2200:H2 = H2 - 1
10130  NEXT H1
10140 H = (C * 70 + 2):V = 25 + V1: GOSUB
    1200
10150 :T$ = "PenMode " + TXT$(C + 1): GOS
   UB 1300
```

```
10160  NEXT C
10170 :
10200  FOR I = 1 TO 4:
10210 V1 = 0:H1 = 0:V2 = 200:H2 = 3200: G
   OSUB 2100
10220  NEXT
10299 :
10300 C = 0: GOSUB 1400
10310  FOR V1 = 0 TO 85 STEP 5
10320 C = V1 / 5: GOSUB 1600
10330 V2 = 200 - V1:H1 = V1:H2 = 320 - H1
   : GOSUB 2200
10340  NEXT V1
10499 :
10500 C = 10: GOSUB 1600
10510 C = 2: GOSUB 1400
10520  FOR S = 0 TO 359 STEP 10
10530 C = 11: GOSUB 2350
10540  GOSUB 2370:
10550  NEXT S:
10600 S = 0:C = 08: GOSUB 1600:C = 0: GOS
   UB 1400
10610  FOR C = 0 TO 360 STEP 9
10620  GOSUB 2350
10630  NEXT
10640  FOR I = 1 TO 3:V1 = 0:H1 = 0:V2 =
   200:H2 = 320: GOSUB 2100: NEXT
10697 :
10698 :
10699 REM exemple d'utilisation de
   CARTOUCHE
10700 C = 5: GOSUB 1600:
10710 : GOSUB 2200
10711 :T$ = "Cartouche"
10798 :
10799  FOR FO = 0 TO 15:C = FO: GOSUB 240
   0:
10800  FOR C = 0 TO 15
10810 V1 = 2 + C * 12:H1 = 30:V2 = V1 + 1
   2:H2 = H1 + 100:
10820 : GOSUB 5000: REM  --- CARTOUCHE
   gauche ---
10821 : GOSUB 5100
10830 C = C + 1
10840 V1 = 6 + (C - 1) * 12:H1 = 170:V2 =
    V1 + 12:H2 = H1 + 100:
10850 : GOSUB 5000: REM  --- CARTOUCHE
   droite ---
10851 : GOSUB 5100:
10860  NEXT C:
10870  NEXT FO
10890 :
10899 : REM ---- exemple d'utilation
   de Centertexte ---
10900 :C = 12: GOSUB 1600:V1 = 0:H1 = 0:V
   2 = 200:H2 = 320: GOSUB 2200
10910 V1 = 090:V2 = V1 + 20::H1 = 090:H2
   = H1 + 150:C = 4: GOSUB 5000
10915 C = 11: GOSUB 2400
10920 T$ = "Couleur": GOSUB 5100
10921 :
10940  FOR CO = 15 TO 0 STEP  - 1
10950 C = CO: GOSUB 1900: GOSUB 5100
10960  NEXT CO
11000 :
11010  FOR TM = 0 TO 02:REM TextMode
11015 C = 04: GOSUB 5000:C = 11: GOSUB 24
   00
11020 C = TM: GOSUB 1700:T$ = "Texte Styl
   e" +  STR$ (TM)
11040  GOSUB 5100
11050  FOR I = 0 TO 600: NEXT I
11090  NEXT TM
11095 T$ = "LineTo sample": GOSUB 5100
11099 :
11100 :REM ==== LineTo sample ====
11110 C = 0:V1 = 0:H1 = 0:V2 = 200:H2 = 3
   20: GOSUB 2300
11120 :
11125  FOR MODE = 3 TO 0 STEP  - 1:C = MO
   DE: GOSUB 1400
11130  FOR I = 0 TO 200 STEP 2:
11140 V = I:H = 0: GOSUB 1200
11150 C =  ABS (15 - I / 13): GOSUB 1600
11160 V = 0:H = 320 - I * 1.6: GOSUB 2000
11170 V = 200 - I::H = 320: GOSUB 2000
11180 V = 200::H = I * 1.6: GOSUB 2000
11185 V = I::H = 0: GOSUB 2000
11190  NEXT I
11192  NEXT MODE
11193 :
11194 V1 = 090:V2 = V1 + 20::H1 = 090:H2
   = H1 + 150:C = 3: GOSUB 5000
11195 :T$ = "The End":C = 12: GOSUB 2400:
    GOSUB 5100
11199 :
19999 :
20000  GET A$: GOSUB 1100
```

Récapitulation QD.BIN

Après avoir saisi cette récapitulation sous moniteur, vous la sauvegarderez par : BSAVE QD.BIN, A$BOO, L597

```
0B00:4C FD 0B 4C 50 0C 4C 64
0B08:0C 4C 79 0C 4C 8C 0C 4C
0B10:9D 0C 4C AA 0C 4C BB 0C
0B18:4C CC 0C 4C DD 0C 4C EE
0B20:0C 4C 03 0D 4C 16 0D 4C
0B28:29 0D 4C 44 0D 4C CF 0B
0B30:00 00 05 00 01 00 00 00
0B38:0A 00 0A 00 64 00 64 00
0B40:00 00 0A 00 14 00 14 00
0B48:0A 68 65 6C 6C 6F 20 30
0B50:31 32 33 00 00 00 00 00
0B58:00 00 00 00 00 00 00 00
0B60:00 00 00 00 00 00 00 00
0B68:00 00 00 00 00 00 00 00
0B70:00 00 00 00 00 00 00 00
0B78:00 00 00 00 00 00 00 00
0B80:00 00 00 00 00 00 00 00
0B88:00 00 00 00 00 00 00 00
0B90:00 00 00 00 00 00 00 00
0B98:00 00 00 00 00 00 00 00
0BA0:00 00 00 00 00 00 00 00
0BA8:00 00 00 00 00 00 00 00
0BB0:00 00 00 00 00 00 00 00
0BB8:00 00 00 00 00 00 00 00
0BC0:00 00 00 00 00 00 00 00
0BC8:00 00 00 EA EA EA EA 60
0BD0:E2 30 38 FB 60 18 FB C2
0BD8:30 60 0B F4 00 00 F4 00
0BE0:00 F4 00 00 F4 00 08 A2
0BE8:02 1A 22 00 00 E1 3B 5B
```

```
0BF0:A0 06 00 B7 01 8F 30 0B
0BF8:00 68 68 2B 60 20 D5 0B
0C00:F4 00 00 A2 04 06 22 00
0C08:00 E1 68 D0 39 20 DA 0B
0C10:F4 00 00 F4 00 00 F4 E1
0C18:00 F4 00 20 A2 02 1A 22
0C20:00 00 E1 A2 02 10 22 00
0C28:00 E1 F4 00 08 F4 00 00
0C30:F4 A0 00 AF 30 0B 00 48
0C38:A2 04 02 22 00 00 E1 90
0C40:02 00 FF 4C D0 0B A2 04
0C48:0A 22 00 00 E1 4C D0 0B
0C50:20 D5 0B A2 04 0B 22 00
0C58:00 E1 A2 04 03 22 00 00
0C60:E1 4C D0 0B 20 D5 0B AD
0C68:44 0B 48 AD 46 0B 48 A2
0C70:04 3A 22 00 00 E1 4C D0
0C78:0B 20 D5 0B F4 00 00 F4
0C80:48 0B A2 04 A5 22 00 00
0C88:E1 4C D0 0B 20 D5 0B AD
0C90:32 0B 48 A2 04 2E 22 00
0C98:00 E1 4C D0 0B 20 D5 0B
0CA0:A2 04 0B 22 00 00 E1 4C
0CA8:D0 0B 20 D5 0B AD 32 0B
0CB0:48 A2 04 37 22 00 00 E1
0CB8:4C D0 0B 20 D5 0B AD 34
0CC0:0B 48 A2 04 9A 22 00 00
0CC8:E1 4C D0 0B 20 D5 0B AD
0CD0:36 0B 48 A2 04 9C 22 00
0CD8:00 E1 4C D0 0B 20 D5 0B
0CE0:AD 32 0B 48 A2 04 A0 22
0CE8:00 00 E1 4C D0 0B 20 D5
0CF0:0B AD 44 0B 48 AD 46 0B
0CF8:48 A2 04 3C 22 00 00 E1
0D00:4C D0 0B 20 D5 0B F4 00
0D08:00 F4 38 0B A2 04 56 22
0D10:00 00 E1 4C D0 0B 20 D5
0D18:0B F4 00 00 F4 38 0B A2
0D20:04 54 22 00 00 E1 4C D0
0D28:0B 20 D5 0B F4 00 00 F4
0D30:38 0B AD 40 0B 48 AD 42
0D38:0B 48 A2 04 63 22 00 00
0D40:E1 4C D0 0B 20 D5 0B AD
0D48:32 0B 48 A2 04 A2 22 00
0050:00 E1 4C D0 0B
```

Heure & AppleWriter

Lyda Pátková

Il ne s'agit pas d'un accessoire de bureau type GS mais plus modestement d'un programme WPL qui permet l'exploitation de la carte horloge depuis votre traitement de textes préféré. Si vous avez une carte Speedisk, vous saurez l'heure actuelle, sinon vous connaîtrez une heure vieille de 3 ou 4 secondes.

Le mode d'emploi est simple : faire :

`<ctrl-P>DO HEURE`

L'heure, sous la forme 10:58 est insérée à l'emplacement du curseur. Faites à nouveau :

`<ctrl-P>DO HEURE`

l'heure est effacée.

Seule contrainte : il faut avoir dans le catalogue courant un sous-catalogue nommé 'N' réservé au seul usage de la récupération de l'heure. Ce sous-catalogue contiendra un fichier composé de '**' uniquement.

Bien sûr, il sera plus confortable de mettre l'ordre WPL dans une commande glossaire : vous aurez l'heure sous la pomme.

Fichier HEURE

][+
//e
//e+
//c
][gs

•PND	*Stoppe l'affichage.*
•PCS/$a/*/	*Si le pointeur $A est *,*
•PGO*	*aller à l'étiquette *.*
•F//**/	*Met dans le texte le repère **
•O?	*à la place du curseur.*
•B	*Début page texte et direction > ligne de statut.*
•F/**/	*Retourne au repère **.*
•?	
•SN/N,*	*Sauvegarde, dans le sous-catalogue N, le fichier N*
•O	*ne contenant que le repère **.*
•UU	*2 flèches de droite : passe devant le repère **
•OAN£	*Charge le catalogue en mémoire texte.*
•?	
•D	*Change la direction ligne de statut : <.*
•F!$$$$2££!**!	*Remplace la fin du catalogue jusqu'à l'heure de*
•O?	*modification par un nouveau repère **.*
•WW	*Met dans le buffer texte l'heure et un espace.*
•UUU	*Déplace le cureur vers la droite, après le repère **.*
•F!**"**!	*Supprime tout ce qu'il y a entre les repères,*
•O?	*repères inclus.*
•D	*Change la direction ligne de statut : >.*
•WW	*Remet l'heure et l'espace du buffer dans le texte.*
•PAS*=$A	*Initialise le pointeur $A.*
•PQT	*Fin d'un premier appel.*
* •D	*Second appel : change la direction.*
•WWU	*Supprime l'heure.*
•PAS=$A	*Vide le pointeur $A.*

ProDOS

Pascal Cantot

ONLINE : une commande externe

Source ONLINE.S
Assembleur Merlin Pro

ProDOS

```
BASIC    KBD

CNTR     =     $18

IN       =     $200
BUFFER   =     $200        ;Utilisons IN comme buffer

         DO    BASIC
EXTRNCMD =     $BE06       ;Vecteur vers commande externe
ERROUT   =     $BE09       ;Traitement d'erreur
XTRNADDR =     $BE50       ;Adresse de la commande externe
XLEN     =     $BE52       ;longueur - 1 de la commande
XCNUM    =     $BE53       ;N° commande BASIC.SYSTEM (ext.=0)
PBITS    =     $BE54       ;Drapeaux des paramètres requis
BADCALL  =     $BE8B       ;Convertit erreur MLI -> Basic
SONLINE  =     $BEC6       ;Liste parametres MLI pour ON_LINE:
SUNITNUM =     $BEC7       ;  Code de l'unité (00 = toutes)
SBUFADR  =     $BEC8       ;  Adresse du buffer
         FIN
MLI      =     $BF00       ;Point d'entrée du MLI ProDOS
SERR     =     $BF0F       ;Code erreur MLI (00 = ok)
BITMAP   =     $BF58       ;Schéma occupation mémoire
PRBYTE   =     $FDDA       ;Affiche A en hexa
COUT     =     $FDED       ;Affiche le caractère dans A
RTS      =     $FF58       ;RTS connu en ROM

On_Line  EQU   $C5         ;Référence de la routine On_Line

*********************************
*
* Code destiné à e ^tre relogé avec CMDLOAD (cf POM'S 20)
* syntaxe commande: "ONLINE", pas de paramètres
* Il faut charger CMDLOAD en $2000, ONLINE.CODE en $2100
* et faire CALL 8192 pour installer la commande ProDOS
*
         ORG   $2100       ;Adresse arbitraire ($2100 pour CMDLOAD)

         DO    BASIC

START    CLD               ;Convention avec ProDOS
         LDA   £>PRGEND+$100 ;Convention avec CMDLOAD
         LDA   £>PRGLEN+$100
V_OLDCMD LDA   RTS         ;Adresse commande précédente

         SEI               ;Inhnibe les interruptions
ONLINCOM LDX   £0          ;Verifie qu'il s'agit bien
CHKCOM   LDA   IN,X        ;de notre commande
```

Qui n'a jamais souhaité disposer sous ProDOS d'une commande donnant la liste de tous les volumes en ligne, comme le font AppleWriter, Merlin-Pro, etc. ? Ne serait-ce que pour connaître le nom de telle disquette, ou savoir si telle autre disquette est bien au format ProDOS...

Cette fonction existe au sein du noyau ProDOS : elle s'appelle `ON_LINE`. Mais elle n'est accessible que via le MLI (Machine Language Interface) qui, comme son nom l'indique, ne peut être appelé que par un programme machine.

La routine ONLINE proposée ici affiche la liste de tous les volumes en précisant port, lecteur et nom d'accès de chacun d'entre eux. De plus, si une erreur survient lors de la lecture d'un volume, le programme affiche à la place du nom le code de l'erreur MLI spécifique à l'unité ayant provoquée l'erreur. Exemple :

```
S6,D1: /MERLIN
S6,D2: $52
S3,D2: /RAM
```

Nous voyons que la disquette (ou plutôt le volume) dans le lecteur 2, port 6 n'est pas un volume ProDOS (c'est peut-être une disquette DOS 3.3, CP/M...). Les erreurs possibles sont les suivantes :

$27 : erreur d'entrée/sortie (I/O ERROR) ;

$28 : pas de périphérique connecté (NO DEVICE CONNECTED) ;

```
        ORA   £$80        ;Convertit en Ascii négatif
        CMP   CMDSTR,X
        BNE   BADCOM      ;Si pas notre commande
        INX
        CPX   £6          ;6 caractères correspondant?
        BCC   CHKCOM      ;Non -> on continue

        CLI               ;Autorise à nouveau les interruptions
        DEX
        STX   XLEN
        LDA   V_ONLINE+1  ;C'est bien "ONLINE"!
        STA   XTRNADDR    ;alors on donne au BASIC.SYSTEM
        LDA   V_ONLINE+2  ;l'ordre de l'exécuter
        STA   XTRNADDR+1
        LDA   £0          ;Rem: on n'indique pas dans PBITS
        STA   SERR        ;qu'aucun paramètre (,S,D,L...)
        STA   XCNUM       ;n'est requis par ONLINE.
        STA   PBITS       ;Donc ONLINE,S6,D1 ne provoquera
        STA   PBITS+1     ;pas d'erreur mais seul "ONLINE"
        CLC               ;sera pris en compte
        RTS               ;Retourne vers le BASIC.SYSTEM

V_ONLINE JMP  ONLINE      ;Vecteur vers ONLINE

BADCOM  SEC               ;Si ce n'est pas "ONLINE"
        CLI               ;Retourne avec C=1
        JMP   (V_OLDCMD+1) ;et essaye comm. suivante

        FIN

* Routine principale: 2 points d'entrées
* ONLINE: liste tous les volumes
* ONLINO: affiche le nom du volume dans l'unité
* dont le code DSSS0000 est dans A
* (D: drive 0-1, SSS: Slot 0-7)

ONLINE  LDA   £0          ;Balaye tous les volumes
ONLINO  STA   SUNITNUM    ;A = dsss0000 (d: drv, sss: slot)
        LDX   £<BUFFER
        STX   SBUFADR
        LDX   £>BUFFER
        STX   SBUFADR+1
        LDA   BITMAP
        AND   £%11011111  ;S'assure que la page 2 (IN)
        STA   BITMAP      ;est marquée comme "libre"

        JSR   MLI         ;Exécute la routine On_Line
        DFB   On_Line     ;du MLI ProDOS
PRMADR  DA    SONLINE
        BEQ   ONLINZ      ;Z=1 si pas d'erreur

        DO    BASIC
        JSR   BADCALL     ;Sinon, affiche le message
        JMP   ERROUT      ;d'erreur (ou gère ONERR)
        ELSE
        RTS               ;Sort avec Z=0, C=1 et A= erreur
        FIN

ONLINZ  LDX   £0
        LDA   CNTR
        PHA
SCANVOL LDA   BUFFER,X    ;Saisie le premier octet
        BEQ   EOBUF       ;Si 00, alors fin
        PHA
        LDA   £$8D        ;Sinon, nouveau volume
```

$2E : disques échangés : fichier encore ouvert sur un autre disque ;
$45 : impossible de trouver le répertoire du volume (VOLUME DIRECTORY NOT FOUND) ;
$52 : ce n'est pas un disque ProDOS (NOT A PRODOS DISK) ;
$57 : nom de volume déjà utilisé (DUPLICATE VOLUME).

ONLINE peut reconnaître jusqu'à 16 volumes (c'est le maximum toléré par ProDOS). La routine utilise le buffer clavier ($200-$2FF) comme espace de travail. Aucune adresse page zéro n'est modifiée. Enfin, il est intéressant de savoir que ONLINE prend moins de 256 octets en mémoire...

Lorsque ONLINE.S est assemblé (par MERLIN PRO), le système demande de donner une valeur à l'étiquette (label) "BASIC" ("Give value for BASIC:"). Si on entre "1", le code objet produit sera celui d'une commande externe ProDOS nommée "ONLINE", installée par CMDLOAD (voir la revue Pom's 20). Si on répond "0" en revanche, le code produit est celui d'une simple routine machine appelée par CALL, JSR, BRUN, etc.

La commande OnLine

Pour l'installer, il suffit de faire :

- *ONLINE.EXE*

(Ce fichier a été obtenu en faisant :

```
BLOAD ONLINE.CODE
BLOAD CMDLOAD
BSAVE ONLINE.EXE,A$2000,L512
```

ONLINE.CODE est le résultat de l'assemblage de ONLINE.S option Basic, CMDLOAD est le relogeur d'A. Avrane (Pom's 20).)

La routine sera relogée sous le BASIC.SYSTEM et la commande sera active et utilisable comme n'importe quelle autre commande de ProDOS.

La syntaxe est :

ONLINE

```
        JSR   COUT        ;Envoie un retour chariot
        LDA   £"S"        ;Affiche "Ss,Dd: "
        JSR   COUT
        PLA
        PHA
        AND   £%01110000  ;Isole le slot
        LSR
        LSR
        LSR
        LSR
        CLC
        ADC   £"0"        ;Convertit en chiffre ascii
        JSR   COUT
        LDA   £","
        JSR   COUT
        LDA   £"D"
        JSR   COUT
        LDY   £0
        PLA
        ASL               ;Bit7 (drive) -> C
        TYA               ;Si C=0, alors drive 1 sinon drive 2
        ROL               ;C -> Bit 0 de A, et bit 7 de A -> C
        ADC   £"1"        ;mais bit 7 = 0, donc C = 0
        JSR   COUT
        LDA   £":"
        JSR   COUT
        LDA   £" "
        JSR   COUT

        LDA   BUFFER,X
        INX
        AND   £$0F        ;Isole la longueur du nom
        PHA
        BEQ   VOLERR      ;Si NameLength = 0, erreur
        STA   CNTR        ;Sinon, on peut afficher le nom
        LDA   £"/"
        JSR   COUT
PRNAME  LDA   BUFFER,X    ;Affiche le nom du volume
        ORA   £$80
        JSR   COUT
        INX
        DEC   CNTR
        BNE   PRNAME

SETDX   PLA
        EOR   £$0F        ;Calcule l'adresse
        STA   CNTR        ;du nom de volume suivant
        TXA               ;= adr. courante + (15-long. nom)
        CLC
        ADC   CNTR
        TAX

        BCC   SCANVOL     ;Toujours pris si moins de 16 volumes

EOBUF   LDA   £$8D        ;On a liste tous les volumes
        JSR   COUT        ;on saute deux lignes
        LDA   £$8D
        JSR   COUT
        PLA               ;On restaure CNTR
        STA   CNTR
        CLC               ;et on sort avec C=1
        LDA   £0          ;Z=1 et A = 0
        RTS

VOLERR  TXA               ;Affiche le code de l'erreur
```

]][+
//e
//e+
//c
]][gs

Tout ce qui suit (paramètres, S, D...) est ignoré par l'interpréteur du Basic.System.

OnLine sans le Basic

Correspond à l'option d'assemblage "BASIC = 0" (fichier *ONLINE.OBJ* non listé ici). La routine s'exécute à l'adresse "adr" indiquée par la directive "ORG" de l'assembleur (par défaut ici $2100).

Un JSR adr (ou CALL adr) affiche la liste de tous les volumes connectés, alors qu'un JSR adr+3 sélectionne un lecteur de disque particulier : en ce cas, mettre le code de l'unité dans l'accumulateur sous la forme DSSS0000, avec D = numéro du lecteur moins un (donc 0 ou 1) et SSS = slot (0-7).

Si vous souhaitez exécuter ONLINE à une autre adresse 'adr' sans réassembler le source, il suffit de mettre en adr+$04, adr+$09 et adr+$0E le poids FORT de la nouvelle adresse d'exécution (cela suppose que la routine se trouve en début de page, donc à une adresse de la forme $xx00, où xx est le poids fort de 'adr').

InterPom's
version 2.0

Des transmissions intelligentes entre :
Apple//™
et/ou
Macintosh™
et/ou
IBM®et compatibles

Bon de commande page 75

Récapitulation ONLINE.EXE

Après avoir saisi cette récapitulation sous moniteur, vous la sauvegarderez par : BSAVE ONLINE.EXE, A$2000,L512

```
2000:AD 00 BF C9 4C F0 05 A9
2008:87 4C ED FD AD 4D BE F0
2010:05 A9 15 4C 09 BE AD 04
2018:21 69 00 20 98 20 90 05
2020:A9 0E 4C 09 BE CD 02 21
2028:90 F6 AE 08 BE 8D 08 BE
2030:8E 07 21 AE 07 BE 8E 06
2038:21 A0 00 8C 07 BE 48 E9
2040:21 85 3C 68 38 E9 04 85
2048:74 A9 21 85 49 84 48 A0
2050:00 B1 48 F0 27 20 8E F8
2058:A4 2F C0 02 D0 0F B1 48
2060:C9 21 90 09 CD 02 21 B0
2068:04 65 3C 91 48 A5 48 38
2070:65 2F 85 48 A5 49 69 00
2078:85 49 D0 D3 A0 00 A9 21
2080:84 3C 85 3D 18 6D 04 21
2088:84 42 88 84 3E 85 3F AD
2090:08 BE 85 43 C8 4C 2C FE
2098:8D FB 20 A5 74 18 69 04
20A0:8D FC 20 86 3D CE FC 20
20A8:F0 47 AD FC 20 8D FD 20
20B0:AD FD 20 48 4A 4A 4A AA
20B8:68 29 07 A8 B9 F3 20 3D
20C0:58 BF D0 E1 A5 3D D0 09
20C8:B9 F3 20 1D 58 BF 9D 58
20D0:BF AD FC 20 38 CE FD 20
20D8:ED FD 20 C0 FB 20 D0 D0
20E0:A5 3D D0 07 18 AE FD 20
20E8:E8 8A 60 A9 00 85 3D F0
20F0:B9 38 60 80 40 20 10 08
20F8:04 02 01 00 00 00 AD 99
```

```
        PHA             ;survenue durant la lecture
        LDA   £"$"      ;du volume
        JSR   COUT
        LDA   BUFFER,X  ;(code MLI hexa)
        JSR   PRBYTE
        PLA
        TAX
        CLC             ;... Et continue!
        BCC   SETDX     ;Toujours pris

        BRK             ;Fin du code à reloger (CMDLOAD)

*******************************


        DO    BASIC
CMDSTR  ASC   "ONLINE" ;Nom de la commande
PRGEND  =     *
PRGLEN  =     PRGEND-START+$100
        ELSE
SONLINE DFB   2         ;Si pas de BASIC.SYSTEM
SUNITNUM DFB  00        ;alors il faut notre propre
SBUFADR DA    BUFFER    ;ParmList
        FIN
```

```
2100:D8 A9 22 A9 02 AD 58 FF
2108:78 A2 00 BD 00 02 09 80
2110:DD F7 21 D0 29 E8 E0 06
2118:90 F1 58 CA 8E 52 BE AD
2120:3C 21 8D 50 BE AD 3D 21
2128:8D 51 BE A9 00 8D 0F BF
2130:8D 53 BE 8D 54 BE 8D 55
2138:BE 18 60 4C 43 21 38 58
2140:6C 06 21 A9 00 8D C7 BE
2148:A2 00 8E C8 BE A2 02 8E
2150:C9 BE AD 58 BF 29 DF 8D
2158:58 BF 20 00 BF C5 C6 BE
2160:F0 06 20 8B BE 4C 09 BE
2168:A2 00 A5 18 48 BD 00 02
2170:F0 61 48 A9 8D 20 ED FD
2178:A9 D3 20 ED FD 68 48 29
2180:70 4A 4A 4A 4A 18 69 B0
2188:20 ED FD A9 AC 20 ED FD
2190:A9 C4 20 ED FD A0 00 68
2198:0A 98 2A 69 B1 20 ED FD
21A0:A9 BA 20 ED FD A9 A0 20
21A8:ED FD BD 00 02 E8 29 0F
21B0:48 F0 31 85 18 A9 AF 20
21B8:ED FD BD 00 02 09 80 20
21C0:ED FD E8 C6 18 D0 F3 68
21C8:49 0F 85 18 8A 18 65 18
21D0:AA 90 9A A9 8D 20 ED FD
21D8:A9 8D 20 ED FD 68 85 18
21E0:18 A9 00 60 8A 48 A9 A4
21E8:20 ED FD BD 00 02 20 DA
21F0:FD 68 AA 18 90 D1 00 CF
21F8:CE CC C9 CE C5 FD BF 4C
```

Impression avec AppleWriter

Le début d'impression d'un fichier, surtout la *ligne du haut*, est souvent perturbée par des caractères restés dans le buffer d'imprimante, en particulier après l'initialisation – caractères programmables ou autres – ou bien espaces de marge de gauche d'une précédente impression.

Pour résoudre tous les problèmes de ce type, il suffit d'inclure au début de la *ligne du haut* le caractère <ctrl-X> qui a justement pour fonction de vider le buffer d'imprimante. Si vous n'utilisez pas cette ligne, elle peut toutefois ne contenir que <ctrl-X> ou bien par exemple <ctrl-I>80N<ctrl-X><esc>E (configuration de l'interface, vidage du buffer, choix des caractères "ELITE").

Elle peut aussi contenir une ligne complexe initialisée par un fichier texte tel celui présenté ci-dessous.

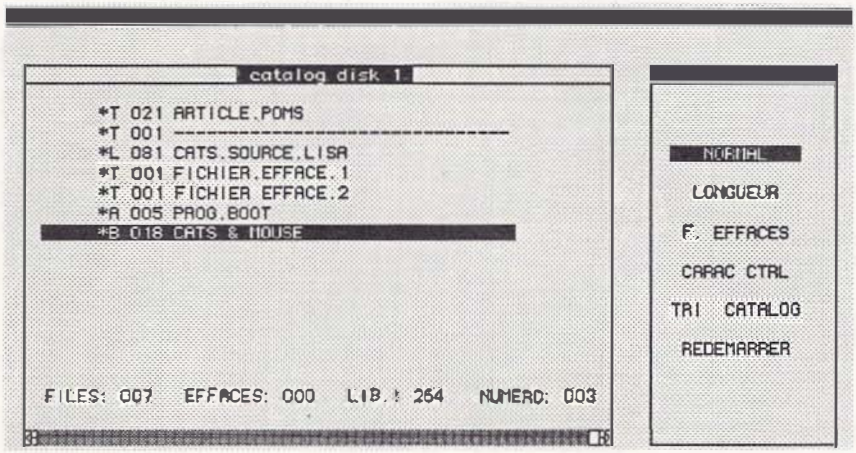
```
.LH*<ctrlI>80N<X><esc>P<esc>!<esc>Xquartier'rastaux' RN 120 MOUSSAILLON 74 550
Caurdes tél : 70.70.70.70     <esc>n<ctrl-N>J.POM'S<esc> q<ctrl-O>    société
nouvelle<esc>E <esc>Y<esc>"*
```

Cats & Mouse

Pascal Duboin

Cet utilitaire permet de visualiser le catalogue d'un disque formaté en DOS 3.3 avec quelques avantages par rapport à la simple commande utilisée par le DOS :

- visualisation des fichiers effacés ;
- visualisation des caractères de contrôle ;
- visualisation des adresses de chargement des fichiers ;
- tri du catalogue (11 options) ;
- défilement de haut en bas et bas en haut du catalogue ;
- utilisation de la souris et de la carte 80 colonnes.



Configuration

Le programme requiert :

- une carte 80 colonnes ;
- le microprocesseur 65C02.

Leur présence est détectée automatiquement en début de programme. En cas de non installation, vous êtes invité à 'rebooter' le système sur un autre disque par simple appui sur la barre d'espacement. L'interface souris n'est pas obligatoire, mais le programme actionne automatiquement l'animal en cas de détection de ladite interface. Le cas échéant, il autorise simplement les ordres entrés au clavier.

Utilisation

Sélections

Dans le menu principal, on sélectionne les options soit en se déplaçant à l'aide des flèches et en validant par *Return*, soit en déplaçant le curseur par un simple 'clic' et en validant par un double 'clic'.

Pour la sélection du lecteur de disques, le principe est le même avec, en plus, la possibilité de s'échapper par *Escape* ou par un simple 'clic' sur le bouton du coin supérieur gauche de la fenêtre.

Ces deux dernières options se retrouvent bien évidemment tout au long du programme.

La suite est aussi classique pour les personnes ayant utilisé un Macintosh. Dans toutes les options d'affichage du catalogue, on peut se déplacer à l'aide des flèches du clavier ou celles dessinées à l'écran, mais vous avez la possibilité de vous déplacer rapidement en frappant un chiffre de 0 à 9, {B} pour Begin, ou encore {E} pour End.

Les utilisateurs de la souris ne seront pas lésés pour autant, car le mode déplacement rapide leur est accessible par simple 'clic' sur la bande de l'ascenseur. Ce mode de déplacement est déconnecté lorsque que l'option *MOVING* est activée.

Le Tri

Dans l'option de *TRI* vous pouvez changer le sens (ascendant ou descendant) par un 'clic' sur le petit bandeau, ou en utilisant la barre d'espacement. Le choix de la méthode de tri est effectué en 'cliquant' sur les chiffres inversés ou en entrant un chiffre au clavier.

Mise à jour

Pour réécrire un catalogue trié, il suffit de 'cliquer' sur le *drive* 1 qui est situé en haut à droite de la fenêtre de travail. Les utilisateurs du clavier se contenteront de taper *W*.

Petites annonces

Vous êtes abonné ? Pom's ouvre gratuitement ses colonnes à vos petites annonces : vente, recherche de matériel ou de logiciel, contact, club... (Pour les ventes de logiciels, joindre la photocopie de la facture d'achat).

Les annonces seront constituées de 250 caractères au maximum et préciseront obligatoirement le type de matériel, le prix et votre numéro de téléphone.

Elles nous seront adressées par courrier ou messagerie CalvaCom (empl1).

Une fenêtre d'alarme s'ouvrira, vous demandant de confirmer votre action soit en cliquant sur *OK*, soit en faisant *Return*. Pour abandonner, il faut cliquer dans tout autre endroit de la fenêtre ou bien encore faire *Escape*.

Cette procédure fonctionne pour toutes les fenêtres, d'alarmes

En ce qui concerne la méthode de tri dite *Personnelle*, elle vous permet de placer vos fichiers dans l'ordre que vous aurez choisi. Pour cela, il convient de sélectionner un fichier à l'aide de l'une des diverses méthodes de déplacement énoncées plus haut et de cliquer sur le *bandeau* inversé, ou encore de frapper *Return*, ce qui aura pour effet de vous mettre en mode *déplacement*.

Le mode 'déplacement'

Ce mode sera caractérisé par l'affichage de *moving* sur la ligne supérieure de la fenêtre de travail. Vous pourrez alors positionner votre fichier à l'emplacement de votre choix, et l'y fixer par un second 'clic' sur le *bandeau* ou *Return*. Pour réécrire le catalogue, la procédure est la même que dans les autres tris, c'est-à-dire : *W* ou 'cliquer' sur *drive : 1* et confirmer par *Return* ou *OK*.

Source 'SOURCE.LISA'

```
        NLS
*
*******************
* THE FRENCH DISK. *
* ---------------- *
*  CATALOG EDITOR  *
*   SORTING MENU   *
*   -------------  *
*   le 25 Mai 87   *
*******************
*
; VARIABLE EN PAGE ZERO
;
DRIVE    EPZ 2
TRACK    EPZ 4
SECTEUR  EPZ 5
DATA     EPZ 8
CMD      EPZ 12
;
INDEX    EPZ 21
LEVEL1   EPZ 22
TSENS    EPZ 23
STYLE    EPZ 24
PTR      EPZ 25
;
CH1      EPZ 27
CV1      EPZ 28
;
LEFT     EPZ 32
RIGHT    EPZ 33
TOP      EPZ 34
DOWN     EPZ 35
CH       EPZ 36
CV       EPZ 37
;
BASL     EPZ 40
STORE    EPZ 44
TEMP     EPZ 46
TEMP1    EPZ 48
;
PDEC     EPZ 68
ICONE    EPZ 81
MSLOT    EPZ 82
;
MAXCLIC  EPZ 84
MINCLIC  EPZ 85
TEMPO    EPZ 86
OPTION   EPZ 88
;
VOIX1    EPZ 89
VOIX2    EPZ 90
LONGEUR  EPZ 91
;
DELFLG   EPZ 92
DELSET   EPZ 93
LGNFLG   EPZ 94
HIDFLG   EPZ 95
SETCUR   EPZ 96
;
ALPHA    EPZ 125
LOCK     EPZ 120
DEL      EPZ 121
WRTOK    EPZ 124
;
FILEOK   EPZ 127
DELETED  EPZ 129
FREE     EPZ 130
FILENBE  EPZ 128
DOS      EPZ 132
FILENUM  EPZ 133
MOVING   EPZ 134
DELCLIC  EPZ 135
;
FAC      EPZ 161
;
; ARITHMETIQUE APPLE
;
INTEGER  EQU $E10C
MULTI    EQU $E982
DIVISE   EQU $EA69
ARG      EQU $EB63
SIGNE    EQU $EB60
FLOAT    EQU $EB93
INT      EQU $EC23
;
; ROM MONITEUR
;
COL80    EQU $C300
BOOTD    EQU $C600
PRAX     EQU $F941
VTAB     EQU $FB5B
TEXT     EQU $FB2F
BASCALC  EQU $FBC1
HOME     EQU $FC58
CROUT    EQU $FC62
IDENT    EQU $FC71
WAIT     EQU $FCA8
PRB      EQU $FDDA
COUT     EQU $FDF6
INVERSE  EQU $FE80
NORMAL   EQU $FE84
SCREEN   EQU $FE93
;
; ADRESSES HARDWARE
;
RBOOT    EQU $3F2
MCH      EQU $478
VMODE    EQU $4FB
MCV      EQU $4F8
MAXL     EQU $4F8
MAXH     EQU $5F8
MINL     EQU $478
MINH     EQU $578
MSTAT    EQU $778
;
LGN10    EQU $528
LGN12    EQU $628
LGN22    EQU $750
LGN23    EQU $700
;
KBD      EQU $C000
STROBE   EQU $C010
CART80   EQU $C017
SPKR     EQU $C030
;
; CONSTANTES
;
SET      =   0
SERVE    =   1
READ     =   2
CLEAR    =   3
POS      =   4
CLAMP    =   5
MHOME    =   6
MINIT    =   7
;
APPLE2C  =   53
APPLE2E  =   06
;
INV      =   $20
CURSOR   =   $42
SABLE    =   $43
FG       =   $88
FB       =   $8A
FH       =   $8B
RTN      =   $8D
CR       =   $8D
FD       =   $95
ESC      =   $9B
SPC      =   $A0
;
BUFFER   EQU $8000
ADRTBL   EQU $8100
ADRSTK   EQU $8200
LENGHT   EQU $9200
START    EQU $9300
TSL      EQU $9400
;
;  OPCODE 6502C
;
INA      =   $1A
DEA      =   $3A
TRB      =   $1C
STZ      =   $64
PHX      =   $DA
PLX      =   $FA
;
         ORG $800
         OBJ $800
;
TEST:
         JSR HOME
         JSR SCREEN
;
         LDA CART80
         BMI TESTERR
         LDA IDENT
         CMP #APPLE2C
         BEQ >0
         CMP #APPLE2E
^0       BEQ BEGIN
;
TESTERR:
         JSR PRINT
         BYT 23,10
         ASC "(PRESS SPACE TO
         BOOT)"
         BRK
;
T0       JSR PRINT
         BYT 10,6
         ASC "< CARTE 80
         COLONNES & 6502c >"
         BRK
;
         LDA #6
         STA TEMPO
         LDA #9
         STA TEMPO+1
;
T1       LDA KBD
         BPL T2
         JMP BOOT2
;
T2       LDA LGN23
         PHA
         LDX #0
         LDY #1
^0       LDA LGN23,Y
         STA LGN23,X
         INX
         INY
         CPY #40
         BCC <0
         PLA
         STA LGN23+39
;
         LDA #215
         JSR WAIT
;
         DEC TEMPO+1
         BPL T1
         LDX #28
         LDA #SPC
^0       STA LGN10+6,X
         DEX
         BPL <0
;
         DEC TEMPO
         BPL T1
         BMI T0
;
BEGIN:
         LDA #BEGIN
         STA RBOOT
         LDA /BEGIN
         STA RBOOT+1
         EOR #$A5
         STA RBOOT+2
;
         JSR COL80
;
         LDX #29
^0       LDA RWTS,X
         STA $0,X
         DEX
         BPL <0
;
         LDX #7
^0       DEC PTR+1
         DEX
         BMI >1
         LDY #12
         LDA (PTR),Y
         CMP #32
         BNE <0
         LDY #251
         LDA (PTR),Y
         CMP #$D6
         BNE <0
         LDA PTR+1
         STA MOUSE+2
         STA MS+2
         EOR #$C0
         STA MSLOT
         BNE MOUSEOK
;
^1       LDA #$60
         STA MSPRT
         STA MOUSEPRT
         STA RDCURS
         BNE TTL
;
MOUSEOK:
         LDX #MINIT
         JSR MOUSE
;
         LDA #79
         STA MAXL
         LDA #I
```

```
        STA MINL
        BYT DEA
        STA MINH
        STA MAXH
        LDX £CLAMP
        JSR MOUSE
;
        LDA £22
        STA MAXL
        LDA £1
        LDX £CLAMP
        JSR MOUSE
;
        LDA £1
        LDX £SET
        JSR MOUSE
        LDX £MHOME
        JSR MOUSE
;
TTL:
;
        JSR PRINT
        BYT 0,255
        ASC ' <- CATS AND MOUSE
        -> '
        ASC 'copyright in june
        1987 by   '
        ASC 'INTERET SOFTWARE &
        DOBOIN.P '
        BRK
;
        JSR HOM3
        JSR WIND
        BYT 1
        BYT 59,20,3,23
        JSR RSCUR
;
INIT:
        BYT ST2,MOVING
        BYT ST2,DBLCLIC
        BYT ST2,ALPHA
        BYT ST2,DEL
        BYT ST2,DELFLG
        BYT ST2,HIDFLG
        BYT ST2,LGNFLG
        LDX £62
        LDA £74
        JSR FIXCLIC
        LDX £255
        STX WRTOK
        TXS
;
        JSR PRINT
        BYT 3,60
        ASC "  CATALOG EDITOR
        "
        BRK
        LDX £17
        JSR REVERT+3
;
MENU:
        JSR PRINT
        BYT 7,62
        ASC "   NORMAL   ^"
        ASC " FILE LENGHT^"
        ASC "FILE DELETED^"
        ASC "HIDDEN TOKEN^"
        ASC "CATALOG SORT^"
        ASC " BOOT DISK  "
        BRK
;
        CLC
        LDA INDEX
        ASL
        ADC £7
        LDX £11
        JSR REVERT
;
        JSR MOUSEPRT
^0      JSR FUNCTION
        BCC MENU
;
        LDX INDEX
        CPX £5
        BEQ >0
        JSR CHOISE
        LDX INDEX
^0      LDA MAINL,X
        STA TEMP
        LDA MAINH,X
        STA TEMP+1
        JMP (TEMP)
;
MAINL:
        BYT CATN
        BYT CATL
        BYT CATD
        BYT CATH
        BYT SORT
        BYT BOOT
;
MAINH:
        HBY CATN
        HBY CATL
        HBY CATD
        HBY CATH
        HBY SORT
        HBY BOOT
;
XMREAD:
        LDX £2
;
MOUSE:
        LDY $C412,X
        STY MS+1
MS      HEX 4C95C4
;
WIND:
        PLA
        STA PTR
        PLA
        STA PTR+1
        LDY £5
W0      LDA (PTR),Y
        STA $1E,Y
        DEY
        BNE W0
;
        LDY "L
        ASL
        BCC >0
        LDY "Ç
^0      STY TEMPO
;
        JSR HOME
        JSR MOUSEON
;
        LDA "Z
        JSR COUT
        LDA RIGHT
        SBC £2
        STA STORE
        TAX
;
        LDA $1F
        BPL C0-2
        LDA "^
        JSR COUT
        DEX
;
        LDA TEMPO
C0      JSR COUT
        DEX
        BPL C0
        LDA "_
        JSR COUT
;
        INC TOP
;
        LDY DOWN
        DEY
        STY STORE+1
        LDX TOP
        LDY RIGHT
        DEY
        STY HERE+1
;
C1      LDA "Z
        JSR COUT
HERE    LDY £0
        STY CH
        LDA "_
        JSR COUT
        INX
        CPX STORE+1
        BNE C1
        DEC TOP
;
        LDX TOP
        LDY DOWN
        DEY
        STY TOP
        JSR MOUSEOFF
        JSR HOME
        JSR MOUSEON
        STX TOP
        INC RIGHT
;
        LDA "Z
        JSR COUT
        JSR MOUSEOFF
        LDA "_
        LDX STORE
C2      JSR COUT
        DEX
        BPL C2
        JSR MOUSEON
        LDA "_
        JSR COUT
        DEC RIGHT
;
        LDA $1F
        BNE C4
;
        LDA TOP
        STA CV
        JSR $FC62
        INC CH
        INC CH
        LDA "Z
        JSR COUT
        LDA "^
        JSR COUT
;
        JSR $FC62
        INC CH
        LDA "S
        LDX STORE
C3      JSR COUT
        DEX
        BPL C3
;
C4      JSR TEXT
        LDA PTR
        ADC £5
        TAX
        LDA PTR+1
        ADC £0
        PHA
        BYT PHX
;
MOUSEOFF:
        JSR NORMAL
        LDA £4
        BNE >0
;
MOUSEON:
        JSR INVERSE
        LDA £1
^0      BYT TRB
        ADR VMODE
        RTS
;
MSPRT:
        LDY MSLOT
        LDA MSTAT,Y
        ASL
        ASL
        BPL MOUSEPRT
RSCUR   JSR MTAB
        JSR ROCURS
;
MOUSEPRT:
        JSR MTAB
        LDA STYLE
        CMP £$1B
        BGE >0
        DEC ICONE
        ORA £$C0
        JSR COUT
        JSR PICKY
        JMP >1
;
^0      JSR COUT
^1      LDY MSLOT
        LDA MCV,Y
        LDX MCH,Y
^0      STA CV1
        STX CH1
        JSR TAB
RDCURS  JSR PICKY
        CMP £CURSOR
        BEQ >0
        CMP £SABLE
        BNE >1
        RTS
^1      STA STYLE
CHGCUR  LDA £CURSOR
^0      JMP COUT
```

Récapitulation 'CAT & MOUSE'

Après avoir saisi cette récapitulation sous moniteur, vous la sauvegarderez par :
BSAVE CAT & MOUSE, A$800, L$1000

```
0800:20 58 FC 20 93 FE AD 17
0808:C0 30 0B AD 71 FC C9 35
0810:F0 02 C9 06 F0 7F 20 91
0818:0C 17 0A A8 D0 D2 C5 D3
0820:D3 A0 D3 D0 C1 C3 C5 A0
0828:D4 CF A0 C2 CF CF D4 A9
0830:00 20 91 0C 0A 06 BC A0
0838:C3 C1 D2 D4 C5 A0 B8 B0
0840:A0 C3 CF CC CF CE CE C5
0848:D3 A0 A6 A0 B6 B5 B0 B2
0850:E3 A0 BE 00 A9 06 85 56
0858:A9 09 85 57 AD 00 C0 10
0860:03 4C BA 0E AD D0 07 48
0868:A2 00 A0 01 B9 D0 07 9D
0870:D0 07 E8 C8 C0 28 90 F4
0878:68 8D F7 07 A9 D7 20 A8
0880:FC C6 57 10 D7 A2 1C A9
0888:A0 9D 2E 05 CA 10 FA C6
0890:56 10 C9 30 9C A9 95 8D
0898:F2 03 A9 08 8D F3 03 49
08A0:A5 8D F4 03 20 00 C3 A2
08A8:1D BD D3 17 95 00 CA 10
08B0:F8 A2 07 C6 1A CA 30 1E
08B8:A0 0C B1 19 C9 20 D0 F3
08C0:A0 FB B1 19 C9 D6 D0 EB
08C8:A5 1A 8D 3E 0A 8D 44 0A
08D0:49 C0 85 52 D0 0D A9 60
08D8:8D 21 08 8D 30 0B 8D 58
08E0:0B D0 33 A2 07 20 3C 0A
08E8:A9 4F 8D F8 04 A9 01 8D
08F0:78 04 3A 8D 78 05 8D F8
08F8:05 A2 05 20 3C 0A A9 16
0900:8D F8 04 A9 01 A2 05 20
0908:3C 0A A9 01 A2 00 20 3C
0910:0A A2 06 20 3C 0A 20 91
0918:0C 00 FF 20 3C 2D 20 43
0920:41 54 53 20 41 4E 44 20
0928:4D 4F 55 53 45 20 2D 3E
0930:20 20 63 6F 70 79 72 69
0938:67 68 74 20 69 6E 20 6A
0940:75 6E 65 20 31 39 38 37
0948:20 62 79 20 20 20 49 4E
0950:54 45 52 45 54 20 53 4F
0958:46 54 57 41 52 45 20 26
0960:20 44 55 42 4F 49 4E 2E
0968:50 20 00 20 8F 0D 20 45
0970:0A 01 3B 14 03 17 20 2A
0978:0B 64 86 64 87 64 7D 64
0980:79 64 5C 64 5F 64 5E A2
0988:3E A9 4A 20 00 0C A2 FF
0990:86 7C 9A 20 91 0C 03 3C
0998:A0 A0 C3 C1 D4 C1 CC CF
09A0:C7 A0 C5 C4 C9 D4 CF D2
09A8:A0 A0 00 A2 11 20 48 0C
09B0:20 91 0C 07 3E A0 A0 A0
09B8:CE CF D2 CD C1 CC A0 A0
09C0:A0 DE A0 C6 C9 CC C5 A0
09C8:CC C5 CE C7 C8 D4 DE C6
09D0:C9 CC C5 A0 C4 C5 CC C5
09D8:D4 C5 C4 DE C8 C9 C4 C4
09E0:C5 CE A0 D4 CF CB C5 CE
09E8:DE C3 C1 D4 C1 CC CF C7
09F0:A0 D3 CF D2 D4 DE A0 C2
09F8:CF CF D4 A0 C4 C9 D3 CB
0A00:A0 A0 00 18 A5 15 0A 69
0A08:07 A2 0B 20 45 0C 20 30
0A10:0B 20 72 0B 90 9A A6 15
0A18:E0 05 F0 05 20 05 0E A6
0A20:15 BD 2E 0A 85 2E BD 34
```

```
;
MTAB:
        LDA CV1
        LDX CH1
        JMP TAB
;
FUNCTION:
        JSR INPUT2
        BCS >3
        LDA STYLE
        CMP £INV+1
        BLT RETURN+4
;
        LDA CV1
        LDX £5
^1      CMP IDX,X
        BEQ >2
        DEX
        BPL <1
        BMI FUNCTION
^2      SBC
        SBC £7
        LSR
        STA INDEX
        LDA STYLE
        AND £$3F
        STA STYLE
        CLC
        RTS
;
IDX     BYT 7,9,11,13,15,17
;
^3      LDX INDEX
        CMP £FD
        BEQ DESC
        CMP £FB
        BNE OTHER
;
DESC    INX
        CPX £6
        BLT >0
        LDX £0
;
^0      STX INDEX
        CLC
        RTS
;
OTHER   CMP £FG
        BEQ MONT
        CMP £FH
        BNE RETURN
;
MONT    DEX
        BPL <0
        LDX £6
        DEX
        BPL <0
;
RETURN  CMP £RTN
        BNE <0
;
CLIC.OK:
        LDA £4
        STA TEMP
        BYT INA
        STA VOIX1
        LSR
        STA VOIX2
        STA LONGEUR
^0      JSR SOUND
        DEC VOIX1
        INC VOIX2
        DEC TEMP
        BNE <0
;
SOUND:
        LDX VOIXI
        LDY VOIX2
        CLC
S1.     DEX
        BNE S2.
        BIT SPKR
        LDX VOIX1
        BCC W1
W1      BCC W2
W2      BCC S2.
S2.     DEY
        BNE S1.
        BIT SPKR
        LDY VOIX2
        DEC LONGEUR
        BNE S1.
^0      SEC
        RTS
;
FIXCLIC:
        STX MINCLIC
        STA MAXCLIC
        BYT ST2,DBLCLIC
        RTS
;
KEY:
        LDA KBD
        BMI >0
        BPL KEY
;
INPUT:
        LDA £255
        STA DBLCLIC
;
INPUT2:
        LDA KBD
        BPL MOUSEMVE
        SEC
^0      STA STROBE
        RTS
;
MOUSEMVE:
        JSR XMREAD
        LDY MSLOT
        LDA MSTAT,Y
        ASL
        ASL
        BPL BOUTONS
;
        JSR MOUSEPRT
;
BOUTONS:
        LDY MSLOT
        LDA MSTAT,Y
        BPL INPUT2
        BIT DBLCLIC
        BMI >1
        ASL
        BMI INPUT2
;
^0      LDX CH1
        CPX MAXCLIC
        BGE INPUT2
        CPX MINCLIC
        BLT INPUT2
;
^1      CLC
        RTS
;
REVERT:
        JSR VTAB
^0      DEC ICONE
        LDY STORE
        JSR COUT2
        INC STORE
        DEX
        BPL <0
        RTS
;
PICKY:
        LDA CV1
        JSR BASCALC
        LDY CH1
;
COUT2:
        TYA
        ROR
        BCS >0
        LDA $C055
        INY
^0      TYA
        LSR
        TAY
        LDA (BASL),Y
        BIT ICONE
        BPL >I
        AND £$3F
        STA (BASL),Y
;
^1      BYT ST2,ICONE
        STA $C054
        RTS
;
AL:
        JSR COUT
        LDA "$
OUT:
        JSR COUT
OUTSPC:
        LDA £SPC
        JMP COUT
;
DRVASC:
        LDA DRIVE
^0      EOR £$B0
        AND £$3F
        RTS
;
TAB:
        JSR VTAB
        STX CH
        RTS
;
PRINT:
        PLA
        STA PTR
        PLA
        STA PTR+1
        LDY £1
;
PR      LDA (PTR),Y
PR0     JSR VTAB
;
PR1     INY
        LDA (PTR),Y
        STA CN
        STA STORE
;
PR2     INY
        LDA (PTR),Y
        BEQ PR3
;
        CMP £255
        BNE >6
        INY
        BNE PR
;
^6      CMP £253
        BNE >8
        INY
        LDA (PTR),Y
        TAX
        INY
        LDA (PTR),Y
^7      JSR COUT
        DEX
        BPL <7
        BMI PR2
;
^8      CMP "^
        BEQ >9
        CMP "^
        BNE >0
        JSR CROUT
^9      JSR CROUT
        LDA STORE
        STA CH
        BPL PR2
;
^0      CMP "!
        BNE >1
        JSR CROUT
        BNE PR1
;
^1      CMP "¢
        BEQ >2
        CMP "£
        BNE >3
;
        INC CV
^2      INC CV
        LDA CV
        BPL PR0
;
^3      CMP £252
        BEQ >4
        CMP £251
        BEQ >5
        JSR COUT
        BNE PR2
;
^4      JSR DRVASC
        STA STR1+1
        LDX £8
^4      LDA STR1,X
        JSR COUT
        DEX
        BPL <4
        BMI PR2
;
^5      LDX £7
^5      LDA STR3,X
        JSR COUT
```

```
0A28:0A 85 2F 6C 2E 00 E3 C8
0A30:DF D9 44 99 0E 0E 0E 0E
0A38:15 0E A2 02 BC 12 C4 8C
0A40:43 0A 4C 95 C4 68 85 19
0A48:68 85 1A A0 05 B1 19 99
0A50:1E 00 88 D0 F8 A0 CC 0A
0A58:90 02 A0 DC 84 56 20 58
0A60:FC 20 18 0B A9 DA 20 F6
0A68:FD A5 21 E9 02 85 2C AA
0A70:A5 1F 10 06 A9 DE 20 F6
0A78:FD CA A5 56 20 F6 FD CA
0A80:10 FA A9 DF 20 F6 FD E6
0A88:22 A4 23 88 84 2D A6 22
0A90:A4 21 88 8C 9C 0A A9 DA
0A98:20 F6 FD A0 00 84 24 A9
0AA0:DF 20 F6 FD E8 E4 2D D0
0AA8:ED C6 22 A6 22 A4 23 88
0AB0:84 22 20 11 0B 20 58 FC
0AB8:20 18 0B 86 22 E6 21 A9
0AC0:DA 20 F6 FD 20 11 0B A9
0AC8:DF A6 2C 20 F6 FD CA 10
0AD0:FA 20 18 0B A9 DF 20 F6
0AD8:FD C6 21 A5 1F D0 24 A5
0AE0:22 85 25 20 62 FC E6 24
0AE8:E6 24 A9 DA 20 F6 FD A9
0AF0:DE 20 F6 FD 20 62 FC E6
0AF8:24 A9 D3 A6 2C 20 F6 FD
0B00:CA 10 FA 20 2F FB A5 19
0B08:69 05 AA A5 1A 69 00 48
0B10:DA 20 84 FE A9 04 D0 05
0B18:20 80 FE A9 01 1C FB 04
0B20:60 A4 52 B9 78 07 0A 0A
0B28:10 06 20 6B 0B 20 58 0B
0B30:20 6B 0B A5 18 C9 1B B0
0B38:0D C6 51 09 C0 20 F6 FD
0B40:20 55 0C 4C 49 0B 20 F6
0B48:FD A4 52 B9 F8 04 BE 78
0B50:04 85 1C 86 1B 20 8B 0C
0B58:20 55 0C C9 42 F0 09 C9
0B60:43 D0 01 60 85 18 A9 42
0B68:4C F6 FD A5 1C A6 1B 4C
0B70:8B 0C 20 12 0C B0 28 A5
0B78:18 C9 21 90 4B A5 1C A2
0B80:05 DD 99 DB F0 05 CA 10
0B88:F8 30 E7 38 E9 07 4A 85
0B90:15 A5 18 29 3F 85 18 18
0B98:60 07 09 0B 0D 0F 11 A6
0BA0:15 C9 95 F0 04 C9 8A D0
0BA8:0B E8 E0 06 90 02 A2 00
0BB0:86 15 18 60 C9 88 F0 04
0BB8:C9 8B D0 08 CA 10 F1 A2
0BC0:06 CA 10 EC C9 8D D0 E8
0BC8:A9 04 85 2E 1A 85 59 4A
0BD0:85 5A 85 58 20 DF 0B C6
0BD8:59 E6 5A C6 2E D0 F5 A6
0BE0:59 A4 5A 18 CA D0 0B 2C
0BE8:30 C0 A6 59 90 00 90 00
0BF0:90 00 88 D0 EF 2C 30 C0
0BF8:A4 5A C6 5B D0 E6 38 60
0C00:86 55 85 54 64 87 60 AD
0C08:00 C0 30 0C 10 F9 A9 FF
0C10:85 87 AD 00 C0 10 05 38
0C18:8D 10 C0 60 20 3A 0A A4
0C20:52 B9 78 07 0A 0A 10 03
0C28:20 30 0B A4 52 B9 78 07
0C30:10 E0 24 87 30 0D 0A 30
0C38:D9 A6 1B E4 54 B0 D3 E4
0C40:55 90 CF 18 60 20 5B FB
0C48:C6 51 A4 2C 20 5C 0C E6
0C50:2C CA 10 F4 60 A5 1C 20
0C58:C1 FB A4 1B 98 6A B0 04
0C60:AD 55 C0 C8 98 4A A8 B1
0C68:28 24 51 10 04 29 3F 91
0C70:28 64 51 8D 54 C0 60 20
0C78:F6 FD A9 A4 20 F6 FD A9
0C80:A0 4C F6 FD A5 02 49 B0
0C88:29 3F 60 20 5B FB 86 24
0C90:60 68 85 19 68 85 1A A0
0C98:01 B1 19 20 5B FB C8 B1
0CAD:19 85 24 85 2C C8 B1 19
```

```
        DEX
        BPL <5
        BIT MSLOT
        BEQ PR3
        JSR MTAB
        LDA £SABLE
        JSR COUT
;
PR3     TYA
        ADC PTR
        TAX
        LDA PTR+1
        ADC £0
        PHA
        BYT PHX
        RTS
;
STR1    ASC '   ;evird'
STR3    ASC "GOLATAC "
;
LS:
        STA CH
        STX PDEC
        TXA
        RTS
LS':
        JSR LS
        JSR DEC255
        JMP NORMAL
DEC255:
        BYT ST2,PDEC+1
        STA PDEC
DEC512:
        LDY £0
^0      LDA PDEC
        SEC
        SBC £100
        STA PDEC
        LDA PDEC+1
        SBC £0
        STA PDEC+1
        BCC >1
        INY
        BNE <0
^1      LDA PDEC
        ADC £100
        STA PDEC
        JSR OUT2
        LDY £0
        LDA PDEC
^2      SEC
        SBC £10
        BCC >3
        STA PDEC
        INY
        BCS <2
^3      JSR OUT2
        LDY PDEC
OUT2    TYA
        AND £15
        ORA £$B0
        JMP COUT
;
HOM3:
        JSR WIND
        BYT 1
        BYT 1,57,3,23
        JMP RSCUR
;
ENTER:
        JSR HOM3
        JMP PRINT
;
QUIT:
        JSR HOM3
        JMP INIT
;
ALARM:
        JSR WIND
        BYT 2
        BYT 11,38,8,16
        JSR PRINT
        BYT 9,22
        BYT 253,14,INV
        BYT 252
        BYT 255,14,41
        ASC ' ok '
        BRK
        JMP PRINT
;
ALRM:
        LDX £13
        LDA £48
;
^0      JSR FIXCLIC
        LDA £0
        LDY £334
        LDX £$9B
        JSR MUSIC
;
^0      JSR RSCUR
        JSR INPUT2
        BCC >4
        CMP £ESC
        BNE >2
^1      JMP QUIT
^2      CMP £RTN
        BNE <0
^3      JSR HOM3
        JMP CLIC.OK
^4      LDA CV1
        CMP £8
        BLT <0
        CMP £15
        BGE <0
        CMP £14
        BNE <1
        LDA CH1
        CMP £41
        BLT <1
        CMP £45
        BGE <1
        BLT <3
;
CHOISE:
        JSR WIND
        BYT 128
        BYT 15,30,8,16
        LDX £16
        LDA £42
        JSR FIXCLIC
;
DK      JSR PRINT
        BYT 8,24
        ASC 'drive select'
        BYT 255,11,22
        ASC " DRIVE NUMBER 1 ^"
        ASC " DRIVE NUMBER 2 "
        BRK
;
        LDA DRIVE
        ASL
        ADC £9
        LDX £15
        JSR REVERT
        JSR MSPRT
;
^0      JSR INPUT2
        BCC>4
        CMP £RTN
        BNE >2
^1      JSR HOM3
        JMP CLIC.OK
^2      CMP £ESC
        BNE >3
^2      JMP QUIT
^3      LDA DRIVE
        EOR £3
        STA DRIVE
        BPL DK
;
^4      LDA STYLE
        CMP '^
        BEQ <2
;
^5      LDX CV1
        CPX £11
        BEQ >6
        CPX £13
        BNE <0
;
^6      LDX CH1
        CPX £22
        BLT <0
        CPX £38
        BGE <0
        CMP '3
        BLT <1
        AND £$3F
        STA STYLE
        BNE <3
;
BOOT:
        JSR ALARM
        BYT 9,14
        ASC ' boot disk'
        BYT 255,12,24
        ASC "VERIFY BOOT"
        BRK
;
        JSR ALRM
BOOT2:
        JSR HOME
        LDA £0
        TAY
        LDX £255
        JSR MUSIC
        JMP BOOTD
;
CATL:
        LDX £0
        TXA
^0      STA LENGHT,X
        STA START,X
        DEX
        BNE <0
        DEX
        STX LGNFLG
        BNE CATN
;
CATH:
        LDA £255
        STA HIDFLG
        BNE CATN
;
CATD:
        LDA £255
        STA DELFLG
;
CATN:
        JSR RDCAT
        BYT ST2,WRTOK
;
        JSR PRTCAT
^0      JSR CATTCH
        BCS <0
        DEC WRTOK
        JMP QUIT
;
CATSECT:
        LDA BUFFER+1
        STA TRACK
        LDA BUFFER+2
        STA SECTEUR
ACESS:
        LDY £0
        TYA
        JSR $B7B5
        BCS ERRDOS
        RTS
;
ERRDOS:
        JSR ALARM
        BYT 9,14
        ASC ' disk error'
        BYT 255,11,19
        ASC "TRACK :""
        ASC "SECTOR:""
        ASC "CMD   :""
        BRK
;
        LDA CMD
        LSR
        BEQ >1
        JSR PRINT
        BYT 13,27
        ASC 'write'
        BRK
        BCC >2
;
^0      INC CV
        LDA CV
        LDX £28
        JSR TAB
        TYA
        JMP PRB
;
^1      JSR PRINT
        BYT 13,27
        ASC "READ"
        BRK
;
^2      LDA £10
        STA CV
        LDY TRACK
        JSR <0
        LDY SECTEUR
        JSR <0
;
```

```
0CA8:F0 7E C9 FF D0 03 C8 D0
0CB0:E8 C9 FD D0 0F C8 B1 19
0CB8:AA C8 B1 19 20 F6 FD CA
0CC0:10 FA 30 E1 C9 FE F0 07
0CC8:C9 DE D0 0C 20 62 FC 20
0CD0:62 FC A5 2C 85 24 10 CD
0CD8:C9 A1 D0 05 20 62 FC D0
0CE0:BD C9 A5 F0 06 C9 A3 D0
0CE8:08 E6 25 E6 25 A5 25 10
0CF0:AA C9 FC F0 09 C9 FB F0
0CF8:18 20 F6 FD D0 A7 20 84
0D00:0C 8D 34 0D A2 08 BD 33
0D08:0D 20 F6 FD CA 10 F7 30
0D10:94 A2 07 BD 3C 0D 20 F6
0D18:FD CA 10 F7 24 52 F0 08
0D20:20 6B 0B A9 43 20 F6 FD
0D28:98 65 19 AA A5 1A 69 00
0D30:48 DA 60 20 20 20 3A 65
0D38:76 69 72 64 C7 CF CC C1
0D40:D4 C1 C3 A0 85 24 86 44
0D48:8A 60 20 44 0D 20 53 0D
0D50:4C 84 FE 64 45 85 44 A0
0D58:00 A5 44 38 E9 64 85 44
0D60:A5 45 E9 00 85 45 90 03
0D68:C8 D0 EE A5 44 69 64 85
0D70:44 20 87 0D A0 00 A5 44
0D78:38 E9 0A 90 05 85 44 C8
0D80:B0 F6 20 87 0D A4 44 98
0D88:29 0F 09 B0 4C F6 FD 20
0D90:45 0A 01 01 39 03 17 4C
0D98:2A 0B 20 8F 0D 4C 91 0C
0DA0:20 8F 0D 4C 79 09 20 45
0DA8:0A 02 0B 26 08 10 20 91
0DB0:0C 09 16 FD 0E 20 FC FF
0DB8:0E 29 20 6F 6B 20 00 4C
0DC0:91 0C A2 0D A9 30 20 00
0DC8:0C A9 00 A0 34 A2 9B 20
0DD0:28 15 20 2A 0B 20 12 0C
0DD8:90 11 C9 9B D0 03 4C A0
0DE0:0D C9 8D D0 F0 20 8F 0D
0DE8:4C C8 0B A5 1C C9 08 90
0DF0:E4 C9 0F B0 E0 C9 0E D0
0DF8:E5 A5 1B C9 29 90 DF C9
0E00:2D B0 DB 90 E0 20 45 0A
0E08:80 0F 1E 08 10 A2 10 A9
0E10:2A 20 00 0C 20 91 0C 08
0E18:18 64 72 69 76 65 20 73
0E20:65 6C 65 63 74 FF 0B 16
0E28:A0 C4 D2 C9 D6 C5 A0 CE
0E30:D5 CD C2 C5 D2 A0 B1 A0
0E38:DE A0 C4 D2 C9 D6 C5 A0
0E40:CE D5 CD C2 C5 D2 A0 B2
0E48:A0 00 A5 02 0A 69 09 A2
0E50:0F 20 45 0C 20 21 0B 20
0E58:12 0C 90 19 C9 8D D0 06
0E60:20 8F 0D 4C C8 0B C9 9B
0E68:D0 03 4C A0 0D A5 02 49
0E70:03 85 02 10 9F A5 18 C9
0E78:5E F0 EF A6 1C E0 0B F0
0E80:04 E0 0D D0 D2 A6 1B E0
0E88:16 90 CC E0 26 B0 C8 C9
0E90:33 90 CD 29 3F 85 18 D0
0E98:D4 20 A6 0D 09 0E 20 62
0EA0:6F 6F 74 20 64 69 73 6B
0EA8:FF 0C 18 D6 C5 D2 C9 C6
0EB0:D9 A0 C2 CF CF D4 00 20
0EB8:C2 0D 20 58 FC A9 00 A8
0EC0:A2 FF 20 28 15 4C 00 C6
0EC8:A2 00 8A 9D 00 92 9D 00
0ED0:93 CA D0 F7 CA 86 5E D0
0ED8:0A A9 FF 85 5F D0 04 A9
0EE0:FF 85 5C 20 EE 0F 64 7C
0EE8:20 D7 11 20 1C 14 B0 FB
0EF0:C6 7C 4C A0 0D AD 01 80
0EF8:85 04 AD 02 80 85 05 A0
0F00:00 98 20 B5 B7 B0 01 60
0F08:20 A6 0D 09 0E 20 64 69
0F10:73 6B 20 65 72 72 6F 72
0F18:FF 0B 13 D4 D2 C1 C3 CB
0F20:A0 BA FE D3 C5 C3 D4 CF
```

```
        JSR ALRM
        JMP INIT
;
WRTCAT:
        JSR ALARM
        BYT 9,14
        ASC ' write catalog'
        BYT 255,12,24
        ASC "ARE YOU SURE"
        BRK
;
        JSR ALRM
;
        JSR PRINT
        BYT 9,21
        ASC "WRITING"
        BYT 251
;
        LDA £$11
        STA TRACK
        STA BUFFER+1
        LDA £16
        STA SECTEUR
        BYT DEA
        STA BUFFER+2
        LDA £255
        STA FILENUM
        INC CMD
;
^0      LDX £11
        DEC SECTEUR
        DEC BUFFER+2
^1      INC FILENUM
        LDA FILENUM
        JSR ADRFIL
        BCS >4
;
        LDY £0
^2      LDA (PTR),Y
        STA BUFFER,X
        INX
        BEQ >3
        INY
        CPY £$23
        BCC <2
        BCS <1
;
^3      JSR ACESS
        JMP <0
;
^4      LDA £0
^5      STA BUFFER,X
        INX
        BNE <5
        JSR ACESS
        JMP QUIT
;
RDCAT:
        JSR PRINT
        BYT 9,22
        ASC "READING "
        BYT 251
;
        LDA £ADRSTK
        STA ADRTBL
        LDA /ADRSTK
        STA ADRTBL+1
        LDX £2
;
^0      LDA ADRTBL-2,X
        CLC
        ADC £35
        STA ADRTBL,X
        LDA ADRTBL-1,X
        ADC £0
        STA ADRTBL+1,X
        INX
        INX
        CPX £254
        BNE <0
        LDA £0
        STA ADRTBL,X
        STA ADRTBL+1,X
;
        LDA £$11
        STA TRACK
        BYT ST2,SECTEUR
        LDA £1
        STA CMD
        LDA /BUFFER
        STA DATA+1
        JSR ACESS
;
        LDA BUFFER+1
        CMP £$11
        BEQ GOODOS
        JSR ALARM
        BYT 9,14
        ASC ' dos error'
        BYT 255,12,21
        ASC "NOT A DOS 3.3
            DISK"
        BRK
        JSR ALRM
        JMP INIT
;
GOODOS:
        BYT ST2,FREE
        BYT ST2,FREE+1
        BYT ST2,FILENBE
        BYT ST2,DELETED
;
        LDY £$8C
DESCR   LDA BUFFER+$38-1,Y
TST     BEQ NXTTRACK
        ASL
        BCC TST
        PHP
        INC FREE
        BNE OTH
        INC FREE+1
OTH     PLP
        BNE TST+2
NXTTRACK DEY
        BNE DESCR
;
        LDA £255
        STA FILENUM
;
^0      JSR CATSECT
        LDX £11
^1      LDA BUFFER,X
        BEQ >3
        BPL NODEL
        INC DELETED
;
NODEL   INC FILENUM
        INC FILENBE
        LDA FILENUM
        JSR ADRFIL
        LDY £0
^2      LDA BUFFER,X
        STA (PTR),Y
        INX
        BEQ <0
        INY
        CPY £35
        BCC <2
        BCS <1
;
^3      LDA FILENUM
        BYT INA
        ASL
        TAY
;
        CLC
        LDA FILENBE
        SBC DELETED
        BYT INA
        STA FILEOK
;
        LDA £0
        STA ADRTBL,Y
        STA ADRTBL+1,Y
        STA FILENUM
        BIT LGNFLG
        BMI RDTSL
^0      RTS
;
RDTSL:
        STA TEMPO
        STA TEMPO+1
        LDA TEMPO
        JSR ADRFIL
        BCS <0
;
        LDY £3
^0      LDA (PTR),Y
        STA BUFFER,Y
        DEY
        BPL <0
;
        CMP £255
        BEQ >4
        STA TRACK
        LDA BUFFER+1
        STA SECTEUR
        LDA BUFFER+2
        JSR FILETYPE
        STA TEMP1
        CMP "T
        BEQ >3
        CMP "?
        BEQ >3
;
        LDA /TSL
        STA DATA+1
        JSR ACESS
        LDA TSL+12
        STA TRACK
        LDA TSL+13
        STA SECTEUR
        JSR ACESS
;
        LDX £3
        LDA TEMP1
        CMP "I
        BGE >2
        CMP "B
        BEQ >2
        LDX £1
^2      LDA TEMPO+1
        ASL
        TAY
        LDA TSL,X
        STA LENGHT+1,Y
        LDA TSL-1,X
        STA LENGHT,Y
        DEX
        DEX
        BMI >3
        LDA TSL,X
        STA START+1,Y
        LDA TSL-1,X
        STA START,Y
;
^3      INC TEMPO+1
        LDA /BUFFER
        STA DATA+1
^4      INC TEMPO
        BNE RDTSL+4
;
ADRFILE:
        TXA
ADRFIL:
        ASL
        TAY
        LDA ADRTBL,Y
        ORA ADRTBL+1,Y
        BEQ >0+1
        LDA ADRTBL,Y
        STA PTR
        LDA ADRTBL+1,Y
        STA PTR+1
        CLC
^0      BIT $38
        RTS
;
ASCENS:
        STA $C055
        LDA 'O
        STA LGN22+1
        BYT INA
        STA LGN22+28
;
        LDX £26
^0      LDA 'V
        STA $C055
        STA LGN22+1,X
        BYT INA
        STA $C054
        STA LGN22+1,X
        DEX
        BNE <0
        STA LGN22+1
;
        LDA £52
        LDX FILENUM
        LDY FILENBE
        DEY
        JSR COMPUTE
        INX
        INX
        INX
        LDA £22
        JSR TAB
        LDA 'C
        JSR COUT
        JMP RSCUR
```

```
0F28:D2 BA FE C3 CD C4 A0 A0
0F30:A0 BA FE 00 A5 0C 4A F0
0F38:1A 20 91 0C 0D 1B 77 72
0F40:69 74 65 00 90 17 E6 25
0F48:A5 25 A2 1C 20 8B 0C 98
0F50:4C DA FD 20 91 0C 0D 1B
0F58:D2 C5 C1 C4 00 A9 0A 85
0F60:25 A4 04 20 46 0F A4 05
0F68:20 46 0F 20 C2 0D 4C 79
0F70:09 20 A6 0D 09 0E 20 77
0F78:72 69 74 65 20 63 61 74
0F80:61 6C 6F 67 FF 0C 18 C1
0F88:D2 C5 A0 D9 CF D5 A0 D3
0F90:D5 D2 C5 00 20 C2 0D 20
0F98:91 0C 09 15 D7 D2 C9 D4
0FA0:C9 CE C7 FB A9 11 85 04
0FA8:8D 01 80 A9 10 85 05 3A
0FB0:8D 02 80 A9 FF 85 85 E6
0FB8:0C A2 0B C6 05 CE 02 80
0FC0:E6 85 A5 85 20 4E 11 B0
0FC8:17 A0 00 B1 19 9D 00 80
0FD0:E8 F0 07 C8 C0 23 90 F3
0FD8:B0 E6 20 FF 0E 4C B9 0F
0FE0:A9 00 9D 00 80 E8 D0 FA
0FE8:20 FF 0E 4C A0 0D 20 91
0FF0:0C 09 16 D2 C5 C1 C4 C9
0FF8:CE C7 A0 FB A9 00 8D 00
1000:81 A9 82 8D 01 81 A2 02
1008:BD FE 80 18 69 23 9D 00
1010:81 BD FF 80 69 00 9D 01
1018:81 E8 E8 E0 FE D0 E9 A9
1020:00 9D 00 81 9D 01 81 A9
1028:11 85 04 64 05 A9 01 85
1030:0C A9 80 85 09 20 FF 0E
1038:AD 01 80 C9 11 F0 28 20
1040:A6 0D 09 0E 20 64 6F 73
1048:20 65 72 72 6F 72 FF 0C
1050:15 CE CF D4 A0 C1 A0 C4
1058:CF D3 A0 B3 AE B3 A0 C4
1060:C9 D3 CB 00 20 C2 0D 4C
1068:79 09 64 82 64 83 64 80
1070:64 81 A0 8C B9 37 80 F0
1078:0D 0A 90 FB 08 E6 82 D0
1080:02 E6 83 28 D0 F3 88 D0
1088:EB A9 FF 85 85 20 F5 0E
1090:A2 0B BD 00 80 F0 1E 10
1098:02 E6 81 E6 85 E6 80 A5
10A0:85 20 4E 11 A0 00 BD 00
10A8:80 91 19 E8 F0 DF C8 C0
10B0:23 90 F3 B0 DD A5 85 1A
10B8:0A A8 18 A5 80 E5 81 1A
10C0:85 7F A9 00 99 00 81 99
10C8:01 81 85 85 24 5E 30 01
10D0:60 85 56 85 57 A5 56 20
10D8:4E 11 B0 F4 A0 03 B1 19
10E0:99 00 80 88 10 F8 C9 FF
10E8:F0 5F 85 04 AD 01 80 85
10F0:05 AD 02 80 20 07 14 85
10F8:30 C9 D4 F0 46 C9 BF F0
1100:42 A9 94 85 09 20 FF 0E
1108:AD 0C 94 85 04 AD 0D 94
1110:85 05 20 FF 0E A2 03 A5
1118:30 C9 C9 80 06 C9 C2 F0
1120:02 A2 01 A5 57 0A A8 BD
1128:00 94 99 01 92 BD FF 93
1130:99 00 92 CA CA 30 0C BD
1138:00 94 99 01 93 BD FF 93
1140:99 00 93 E6 57 A9 80 85
1148:09 E6 56 D0 88 8A 0A A8
1150:B9 00 81 19 01 81 F0 0C
1158:B9 00 81 85 19 B9 01 81
1160:85 1A 18 24 38 60 8D 55
1168:C0 A9 4F 8D 51 07 1A 8D
1170:6C 07 A2 1A A9 56 8D 55
1178:C0 9D 51 07 1A 8D 54 C0
1180:9D 51 07 CA D0 EE 8D 51
1188:07 A9 34 A6 85 A4 80 88
1190:20 A3 11 E8 E8 E8 A9 16
1198:20 88 0C A9 5C 20 F6 FD
11A0:4C 2A 0B 8E 80 11 8C BE
```

```
COMPUTE:
        STX A1+1
        STY A2+1
        JSR FLOAT
        JSR ARG
A1      LDA £0
        JSR FLOAT
        JSR SIGNE
        JSR MULTI
        JSR ARG
A2      LDA £0
        JSR FLOAT
        JSR SIGNE
        JSR DIVISE
        JSR INT
        JSR ARG
        JSR SIGNE
        JSR INTEGER
        LDX FAC
        RTS
;
PRTCAT:
        JSR DRVASC
        STA CATDSK
        JSR ENTER
        BYT 3,2
        ASC '^'
        BYT 253,18,'ç'
        ASC ' catalog disk '
CATDSK  ASC '  '
        BYT 253,18,'ç'
        BRK
;
PRTCAT":
        BIT DELFLG
        BMI >1
        LDA FILEOK
        STA FILENBE
;
^0      LDA DELETED
        PHA
        LDA £$60
        STA PRG1
        LDX £5
        JSR PRG0
        LDA £$20
        STA PRG1
        PLA
        STA DELETED
;
^1      JSR PRINT
        BYT 20,4
        ASC "FILES:"
        BYT 253,6,SPC
        ASC "DELETED: "
        BYT 253,5,SPC
        ASC "FREE:"
        BYT 253,6,SPC
        ASC "NUMBER:"
        BRK
;
        LDX FILEOK
        BNE >0
        JSR INVERSE
^0      LDA £11
        JSR LS'
        LDA £26
        LDX DELETED
        BNE >0
        JSR INVERSE
^0      JSR LS'
;
        LDA FREE+1
        STA PDEC+1
        LDA £38
        LDX FREE
        JSR LS
        JSR DEC$12
;
        LDA FILEOK
        BNE PRTCAT'
        JSR INVERSE
        TAX
        LDA £52
        JSR LS'
        JSR ASCENS
;
        JSR ALARM
        BYT 9,14
        ASC ' catalog'
        BYT 255,12,22
        ASC "NO FILE ON DISK"
        BRK
        JSR ALRM
        JMP INIT
;
PRTCAT':
        BIT MOVING
        BPL >0
        JSR PRINT
        BYT 3,25
        ASC '< moving >'
        BRK
;
^0      LDA £20
        JSR VTAB
        LDA £52
        LDX FILENUM
        INX
        JSR LS'
        JSR ASCENS
        LDA £5
        JSR VTAB
        LDA FILENUM
        SEC
        SBC £8
        BPL >0
        LDA £0
^0      TAX
^1      TXA
        CMP FILENUM
        BNE >2
;
        DEC ICONE
;
^2      JSR ADRFIL
        BCS >3
        JSR PRTFILE
        INX
        INC CV
        LDA CV
        CMP £19
        BCC <1
;
^3      LDA CV
        STA TOP
        LDA £19
        STA DOWN
        LDA £2
        STA LEFT
        LDA £54
        STA RIGHT
        JSR HOME
        JSR $FB2F
        BIT SETCUR
        BMI >0
        JMP MSPRT
^0      JMP RSCUR
;
PRTFILE:
        STX STORE+1
        LDA CV
^0      LDX £4
        STX STORE
        JSR TAB
;
        LDY £0
        STY TEMP
;
        LDA (PTR),Y
        BPL >0
;
        BIT DELFLG
        BMI >8
        DEC CV
        JMP >9
;
^8      DEC TEMP
        LDX £4
^8      LDA STRG1,X
        JSR COUT
        DEX
        BPL <8
        BMI >1
;
STRG1:
        ASC " LED "
;
^0      LDX £4
        BIT LGNFLG
        BPL >8
        JSR $FC10
        JMP >1
^8      JSR OUTSPC
        DEX
        BPL <8
;
^1      LDY £2
        LDA (PTR),Y
        PHA
        BPL >2
        LDA "*
        BNE >3
^2      LDA £SPC
^3      JSR COUT
        PLA
        JSR FILETYPE
        JSR OUT
;
        LDY £$21
        LDA (PTR),Y
        JSR DEC255
        JSR OUTSPC
;
        LDY £3
^5      LDA (PTR),Y
        CMP £SPC
        BCS >7
        CMP £$80
        BLT >7
;
        BIT HIDFLG
        BPL >6
        BIT ICONE
        BMI >6
        EOR £$60
        EOR £$80
        BNE >7
^6      AND £$7F
^7      CMP £$1B
        BGE >7
        ADC £$C0
;
^7      JSR COUT
;
^8      INY
        CPY £32
        BCC <5
;
        LDA (PTR),Y
        BIT TEMP
        BPL >8
        LDA £SPC
^8      JSR OUT
;
        LDX £42
        BIT LGNFLG
        BPL >9
        DEC STORE
        LDA STORE+1
        JSR ADRPRT
        LDX £52
;
^9      BIT ICONE
        BPL >9
        JSR REVERT+3
;
^9      LDX STORE+1
        RTS
;
ADRPRT:
        ASL
        TAY
        LDX £15
        STX TEMP
        LDA START,Y
        ORA START+1,Y
        BEQ >0
;
        LDA "A
        JSR AL
        LDA START+1,Y
        LDX START,Y
        JSR PRAX
        LDA ",
        JSR COUT
        BYT ST2,TEMP
;
^0      LDA LENGHT,Y
        ORA LENGHT+1,Y
        BEQ >1
        LDA "L
        JSR AL
        LDA LENGHT+1,Y
        LDX LENGHT,Y
        JSR PRAX
        BIT TEMP
        BEQ >2
```

```
11A8:11 20 93 EB 20 63 EB A9
11B0:00 20 93 EB 20 80 EB 20
1188:82 E9 20 63 EB A9 00 20
11C0:93 EB 20 80 EB 20 69 EA
11C8:20 23 EC 20 63 EB 20 80
11D0:EB 20 0C E1 A6 A1 60 20
11D8:84 0C 8D F4 11 20 9A 00
11E0:03 02 5E FD 12 5C 20 63
11E8:61 74 61 6C 6F 67 20 64
11F0:69 73 6B 20 20 20 FD 12
11F8:5C 00 24 5C 30 19 A5 7F
1200:85 80 A5 81 48 A9 60 8D
1208:81 17 A2 05 20 7F 16 A9
1210:20 8D 81 17 68 85 81 20
1218:91 0C 14 04 C6 C9 CC C5
1220:D3 BA FD 06 A0 C4 C5 CC
1228:C5 D4 C5 C4 BA A0 FD 05
1230:A0 C6 D2 C5 C5 BA FD 06
1238:A0 CE D5 CD C2 C5 D2 BA
1240:00 A6 7F D0 03 20 80 FE
1248:A9 0B 20 4A 0D A9 1A A6
1250:81 D0 03 20 80 FE 20 4A
1258:0D A5 83 85 45 A9 26 A6
1260:82 20 44 0D 20 57 0D A5
1268:7F D0 32 20 80 FE AA A9
1270:34 20 4A 0D 20 66 11 20
1278:A6 0D 09 0E 20 63 61 74
1280:61 6C 6F 67 FF 0C 16 CE
1288:CF A0 C6 C9 CC C5 A0 CF
1290:CE A0 C4 C9 D3 CB 00 20
1298:C2 0D 4C 79 09 24 86 10
12A0:10 20 91 0C 03 19 3C 20
12A8:6D 6F 76 69 6E 67 20 3E
12B0:00 A9 14 20 5B FB A9 34
12B8:A6 85 E8 20 4A 0D 20 66
12C0:11 A9 05 20 5B FB A5 85
12C8:38 E9 08 10 02 A9 00 AA
12D0:8A C5 85 D0 02 C6 51 20
12D8:4E 11 B0 0C 20 08 13 E8
12E0:E6 25 A5 25 C9 13 90 E8
12E8:A5 25 85 22 A9 13 85 23
12F0:A9 02 85 20 A9 36 85 21
12F8:20 58 FC 20 2F FB 24 60
1300:30 03 4C 21 0B 4C 2A 0B
1308:86 2D A5 25 A2 04 86 2C
1310:20 8B 0C A0 00 84 2E B1
1318:19 10 1D 24 5C 30 05 C6
1320:25 4C AF 13 C6 2E A2 04
1328:BD 33 13 20 F6 FD CA 10
1330:F7 30 17 A0 CC C5 C4 A0
1338:A2 04 24 5E 10 06 20 10
1340:FC 4C 4A 13 20 7F 0C CA
1348:10 FA A0 02 B1 19 48 10
1350:04 A9 AA D0 02 A9 A0 20
1358:F6 FD 68 20 07 14 20 7C
1360:0C A0 21 B1 19 20 53 0D
1368:20 7F 0C A0 03 B1 19 C9
1370:A0 B0 14 C9 B0 90 10 24
1378:5F 10 0A 24 51 30 06 49
1380:60 49 80 D0 02 29 7F C9
1388:1B B0 02 69 C0 20 F6 FD
1390:C8 C0 20 90 D8 B1 19 24
1398:2E 10 02 A9 A0 20 7C 0C
13A0:A2 2A 24 5E 10 09 C6 2C
13A8:A5 2D 20 B9 13 A2 34 24
13B0:51 10 03 20 48 0C A6 2D
13B8:60 0A A8 A2 0F 86 2E B9
13C0:00 93 19 01 93 F0 15 A9
13C8:C1 20 77 0C B9 01 93 BE
13D0:00 93 20 41 F9 A9 AC 20
13D8:F6 FD 64 2E B9 00 92 19
13E0:01 92 F0 14 A9 CC 20 77
13E8:0C B9 01 92 BE 00 92 20
13F0:41 F9 24 2E F0 08 A2 08
13F8:20 7F 0C CA D0 FA 60 D4
1400:C9 C1 C2 03 D2 BF CC 29
1408:7F A0 07 0A 0A B0 03 88
1410:D0 FA B9 FF 13 60 20 9D
1418:12 20 30 0B 20 0E 0C B0
1420:72 A5 18 A6 1C E0 04,90
```

```
        LDX £8
^1      JSR OUTSPC
        DEX
        BNE <1
^2      RTS
;
FILECDE ASC "TIABSR?L"
;
FILETYPE:
        AND £$7F
        LDY £7
        ASL
^0      ASL
        BCS >1
        DEY
        BNE <0
^1      LDA FILECDE,Y
        RTS
;
;
CATTCH':
        JSR PRTCAT'
        JSR MOUSEPRT
CATTCH:
        JSR INPUT
        BCS >1
;
        LDA STYLE
        LDX CV1
        CPX £4
        BLT >8
        CPX £20
        BGE >8
;
        LDA CH1
        CMP £58
        BGE CATTCH
        LDA STYLE
        BMI CATTCH
        CMP £$40
        BCS CATTCH
        LDA £255
        STA TEMP+1
        SEC
        RTS
;
^8      CPX £3
        BNE >8
        LDX CH1
        CPX £48
        BLT >8
        CPX £56
        BCS >8
        BIT WRTOK
        BPL >8
        STA CHGCUR+1
        JSR RSCUR
        LDA £CURSOR
        STA CHGCUR+1
        BYT ST2,WRTOK
        LDA £$C0
        JSR WAIT
        JMP WRTCAT
;
^8      CMP 'O
        BEQ CATUP
        CMP '^
        BEQ >3
        CMP 'W
        BEQ >9
        CMP 'P
        BEQ CATDW
        CMP 'V
        BNE CATTCH
;
^9      LDA CH1
        BYT DEA
        BYT DEA
^9      LDX FILENBE
        DEX
        LDY £52
;
        BIT MOVING
        BMI CATTCH'
        JSR COMPUTE
        STX FILENUM
        BPL CATTCH'
;
^3      JMP QUIT
;
^1      CMP £ESC
        BEQ <3
        CMP £FG
        BEQ CATUP
        CMP £FB
        BEQ CATUP
        CMP £FD
        BEQ CATDW
        CMP £FB
        BEQ CATDW
        CMP "B
        BEQ CATBG
        CMP "E
        BEQ CATEN
        CMP "W
        BNE >0
        BIT WRTOK
        BPL >0
        JMP WRTCAT
;
^0      CMP £RTN
        BEQ >0
        SEC
        SBC £$B0
        BMI >0
        CMP £10
        BCS >0
        TAX
        LDA FILEPOS,X
        BPL <9
^0      SEC
        RTS
;
CATUP:
        JSR LIMH
        BCS >1
        BIT MOVING
        BPL >0
        JSR EXCH2
^0      DEC FILENUM
^1      JMP CATTCH'
;
CATDW:
        JSR LIMB
        BCS <1
        BIT MOVING
        BPL >0
        JSR EXCH1
^0      INC FILENUM
^1      JMP CATTCH'
;
CATBG:
        JSR LIMH
        BCS <1
        BIT MOVING
        BPL >0
        JSR EXCH2
^0      DEC FILENUM
        BPL CATBG
        BMI <1
;
CATEN:
        JSR LIMB
        BCS <1
        BIT MOVING
        BPL >0
        JSR EXCH1
^0      INC FILENUM
        BPL CATEN
        BMI <1
;
LIMH:
        LDA FILENUM
        BEQ LIMERR
        CLC
^0      RTS
;
LIMB:
        LDA FILENUM
        BYT INA
        CMP FILENBE
        BCC <0
;
LIMERR:
        LDA £50
        LDX £230
        LDY £230
;
MUSIC:
        STA LONGEUR
        STX VOIX1
        STY VOIX2
        JMP SOUND
;
NOISE:
        LDA £125
        LDX £20
        LDY £40
        BNE MUSIC
;
FILEPOS BYT 0,6,12,18,24,29,
        35,41,47,52,58
;
SORT:
        JSR WIND
        BYT 1
        BYT 15,30,5,21
        JSR PRINT
        BYT 5,16
        ASC '^ catalog sort'
        BYT 253,4,INV,252
        BYT 255,8,23
        ASC '1'
S1      ASC "- ALPHABETICAL^"
        INV "2"
S2      ASC "- NUMERICAL^"
        INV "3"
S3      ASC "- FILE TYPE^"
        INV "4"
S4      ASC "- FILE LOCK^"
        INV "5"
S5      ASC "- PERSONAL^"
        INV "6"
S6      ASC "- PURGE DELETE"
        BRK
;
        LDX £16
        LDA £44
        JSR FIXCLIC
;
SI:
        BIT TSENS
        BPL ASC
        JSR PRINT
        BYT 20,26
        ASC ' negatif '
        BRK
        BNE >0
;
ASC     JSR PRINT
        BYT 20,26
        ASC ' positif '
        BRK
;
^0      JSR RSCUR
;
^0      JSR INPUT2
        BCS >2
        LDA STYLE
        CMP '^
        BEQ >3
        LDX CV1
        CPX £5
        BEQ <0
        CPX £20
        BNE >1
        CMP "_
        BNE >4
^1      SEC
        SBC '1
        BMI <0
        CMP £7
        BLT SG
;
^2      CMP £SPC
        BEQ >4
        CMP "7
        BGE >2
        CMP "0
        BLT >2
        SEC
        SBC "1
        BPL SG
;
^2      CMP £ESC
        BNE <0
^3      JMP QUIT
;
^4      JSR TSENS.
        JSR NOISE
        BCS SI
;
TSENS.:
        LDA TSENS
        EOR £255
        STA TSENS
        LDA TS2
        LDX TS
        STX TS2
```

```
1428:18 E0 14 B0 14 A5 1B C9
1430:3A B0 E9 A5 18 30 E5 C9
1438:40 B0 E1 A9 FF 85 2F 38
1440:60 E0 03 D0 23 A6 1B E0
1448:30 90 1D E0 3B B0 19 24
1450:7C 10 15 8D 67 0B 20 2A
1458:0B A9 42 8D 67 0B 64 7C
1460:A9 C0 20 A8 FC 4C 71 0F
1468:C9 4F F0 63 C9 5E F0 20
1470:C9 57 F0 08 C9 50 F0 68
1478:C9 56 D0 A0 A5 1B 3A 3A
1480:A6 80 CA A0 34 24 86 30
1488:8D 20 A3 11 B6 85 10 86
1490:4C A0 0D C9 9B F0 F9 C9
1498:88 F0 34 C9 8B F0 30 C9
14A0:95 F0 3D C9 8A F0 39 C9
14A8:C2 F0 46 C9 C5 F0 54 C9
14B0:D7 D0 07 24 7C 10 03 4C
14B8:71 0F C9 8D F0 0F 3B E9
14C0:B0 30 0A C9 0A B0 06 AA
14C8:BD 39 15 10 B3 3B 60 20
14D0:15 15 B0 09 24 86 10 03
14D8:20 38 17 C6 85 4C 16 14
14E0:20 1B 15 B0 F8 24 86 10
14E8:03 20 1D 17 E6 85 4C 16
14F0:14 20 15 15 B0 F8 24 86
14F8:10 03 20 38 17 C6 85 10
1500:F0 30 EB 20 1B 15 B0 E6
1508:24 86 10 03 20 1D 17 E6
1510:85 10 F0 30 D9 A5 85 F0
1518:09 18 60 A5 85 1A C5 80
1520:90 F8 A9 32 A2 E6 A0 E6
1528:85 5B 86 59 84 5A 4C DF
1530:0B A9 7D A2 14 A0 28 D0
1538:EF 00 06 0C 12 18 1D 23
1540:29 2F 34 3A 20 45 0A 01
1548:0F 1E 05 15 20 91 0C 05
1550:10 5E 20 63 61 74 61 6C
1558:6F 67 20 73 6F 72 74 FD
1560:04 20 FC FF 08 17 31 AD
1568:A0 C1 CC D0 C8 C1 C2 C5
1570:D4 C9 C3 C1 CC DE 32 AD
1578:A0 CE D5 CD C5 D2 C9 C3
1580:C1 CC DE 33 AD A0 C6 C9
1588:CC C5 A0 D4 D9 D0 C5 DE
1590:34 AD A0 C6 C9 CC C5 A0
1598:CC CF C3 CB DE 35 AD A0
15A0:D0 C5 D2 D3 CF CE C1 CC
15A8:DE 36 AD A0 D0 D5 D2 C7
15B0:C5 A0 C4 C5 CC C5 D4 C5
15B8:00 A2 10 A9 2C 20 00 0C
15C0:24 17 10 11 20 91 0C 14
15C8:1A 20 6E 65 67 61 74 69
15D0:66 20 00 D0 0F 20 91 0C
15D8:14 1A 20 70 6F 73 69 74
15E0:69 66 20 00 20 2A 0B 20
15E8:12 0C B0 1D A5 18 C9 5E
15F0:F0 2C A6 1C E0 05 F0 EF
15F8:E0 14 D0 04 C9 DF D0 21
1600:38 E9 31 30 E2 C9 07 90
1608:34 C9 A0 F0 14 C9 B7 B0
1610:09 C9 B0 90 05 38 E9 B1
1618:10 23 C9 9B D0 C9 4C A0
1620:0D 20 29 16 20 31 15 B0
1628:97 A5 17 49 FF 85 17 AD
1630:3C 16 AE 0B 17 8E 3C 16
1638:8D 0B 17 60 B0 85 58 20
1640:C8 0B 20 8F 0D A9 FF 85
1648:5C 20 EE 0F A6 58 BD 9E
1650:16 85 19 BD 98 16 85 1A
1658:A0 00 B1 19 F0 15 C9 DE
1660:F0 11 C9 A0 D0 04 A9 20
1668:D0 03 38 E9 60 99 90 17
1670:C8 10 E7 A9 20 C0 0E B0
1678:06 99 90 17 C8 D0 F6 BD
1680:B6 16 8D D6 16 BD B0 16
16B8:8D D8 16 BD AA 16 85 19
1690:BD A4 16 85 1A 6C 19 00
1698:15 15 15 15 15 15 69 79
16A0:86 93 A0 AC 17 17 17 17
```

```
        STA TS
        RTS
;
TS2     ASC "0"
;
SG:
        STA OPTION
        JSR CLIC.OK
        JSR HOM3
        LDA £255
        STA DELFLG
        JSR RDCAT
;
NR      LDX OPTION
        LDA STL,X
        STA PTR
        LDA STH,X
        STA PTR+1
        LDY £0
^3      LDA (PTR),Y
        BEQ >6
        CMP "^"
        BEQ >6
        CMP £SPC
        BNE >4
        LDA £INV
        BNE >5
^4      SEC
        SBC £$60
^5      STA SE+2,Y
        INY
        BPL <3
^6      LDA £INV
^7      CPY £14
        BGE PRG0
        STA SE+2,Y
        INY
        BNE <7
;
PRG0    LDA SS,X
        STA SS2+1
        LDA SL,X
        STA SL2+1
;
        LDA SORTL,X
        STA PTR
        LDA SORTH,X
        STA PTR+1
        JMP (PTR)
;
STH:
        HBY S1+2
        HBY S2+2
        HBY S3+2
        HBY S4+2
        HBY S5+2
        HBY S6+2
;
STL:
        BYT S1+2
        BYT S2+2
        BYT S3+2
        BYT S4+2
        BYT S5+2
        BYT S6+2
;
SORTH:
        HBY ALPH
        HBY NUM
        HBY TYP
        HBY LCK
        HBY PC
        HBY PRG
;
SORTL:
        BYT ALPH
        BYT NUM
        BYT TYP
        BYT LCK
        BYT PC
        BYT PRG
;
SL      BYT 33,34,3,3,0,0
SS      BYT 2,32,1,1,0,255
;
TRI:
        BYT ST2,TEMPO
PA:
        LDX TEMPO
        STX FILENUM
        JSR ADRFILE
        BCS END
        INX
        STX TEMPO
        STA STORE+1
        LDA PTR
        STA STORE
        JSR ADRFILE
        BCS END
;
SS2     LDY £2
SL2     CPY £33
        BEQ PA
        INY
        LDA (STORE),Y
        BIT ALPHA
        BPL >3
        BIT DEL
        BPL >1
        CMP £255
        JMP ST+2
;
^1      BIT LOCK
        BMI >2
        AND £127
        PHA
        LDA (PTR),Y
        AND £127
        STA TEMP
        PLA
        CMP TEMP
        JMP ST
;
^2      BIT LOCK
        BPL >3
        CMP £70
        JMP ST
;
^3      CMP (PTR),Y
ST      BEQ SL2
TS      BLT PA
;
^0      JSR EXCH1
        BEQ PA
;
END:
        INC DOS
        LDA DOS
        CMP FILENUM
        BNE TRI
        BYT ST2,FILENUM
        RTS
;
EXCH1:
        LDA FILENUM
        ASL
        TAX
        LDY £2
^0      LDA ADRTBL,X
        STA TEMPO+1
        LDA ADRTBL+2,X
        STA ADRTBL,X
        LDA TEMPO+1
        STA ADRTBL+2,X
        INX
        DEY
        BNE <0
        RTS
;
EXCH2:
        LDA FILENUM
        ASL
        TAX
        LDY £2
^0      LDA ADRTBL,X
        STA TEMPO+1
        LDA ADRTBL-2,X
        STA ADRTBL,X
        LDA TEMPO+1
        STA ADRTBL-2,X
        INX
        DEY
        BNE <0
        RTS
;
LCK:
        DEC LOCK
        BNE TYP
;
PRG:
        DEC DEL
        BIT TSENS
        BPL TYP
        JSR TSENS.
TYP:
        DEC ALPHA
NUM:
ALPH:
        BYT ST2,DOS
        JSR TRI
        BIT DEL
        BPL >0
        LDA FILEOK
        STA FILENBE
        ASL
        TAY
        LDA £0
        STA DELETED
        STA ADRTBL,Y
        STA ADRTBL+1,Y
;
^0      BYT ST2,DEL
        BYT ST2,LOCK
        BYT ST2,ALPHA
;
PRG1    JSR SENT
^0      JSR CATTCH
        BCS <0
;
SENT:
        JSR ENTER
        BYT 3,2
SE      ASC '^ purge deleted '
SE2     BYT 253,29,INV,252
        BRK
        LDA £255
        STA SEICUR
        JMP PRTCAT"
;
;
PC:
        JSR SENT
        BPL >1
^0      JSR PRTCAT'
^1      BYT ST2,TEMP+1
        JSR CATTCH
;
        BIT TEMP+1
        BMI >2
        CMP £CR
        BNE <0
^2      LDA MOVING
        EOR £255
        STA MOVING
        JSR PRINT
        BYT 3,25
        BYT 253.9,INV
        BRK
        JSR MOISE
        BCS <0
;
;
RWTS:
        ORG $00
        OBJ RWTS
;
        HEX 01
        HEX 60
        HEX 01
        BRK
        BRK
        BRK
        ADR DCT
        ADR BUFFER
        HEX 0000
        HEX 01
        HEX 00006001
DCT     BRK
        HEX 01
        HEX EFD8
;
        HEX 000000A0
        HEX 00C8
        HEX 0100
;
        ORG RWTS+$1D
        OBJ *
        DFS $1800-*,0
;
        LST
;
; PASCAL DOBOIN
; VIGNY THYEZ
; 74300 CLUSES
;
; ASSEMBLEUR: LISA 2.5
; CREATION  : JAN 1987
; MODIFS    : MAI 1987
;
; ADRESSE DEBUT: $800  soit 2048
; ADRESSE FIN  : $17FF soit 6143
; LONGEUR      : $FFF  soit 4095
; TAILLE DISK  : 18 SECTEURS
;
        END
```

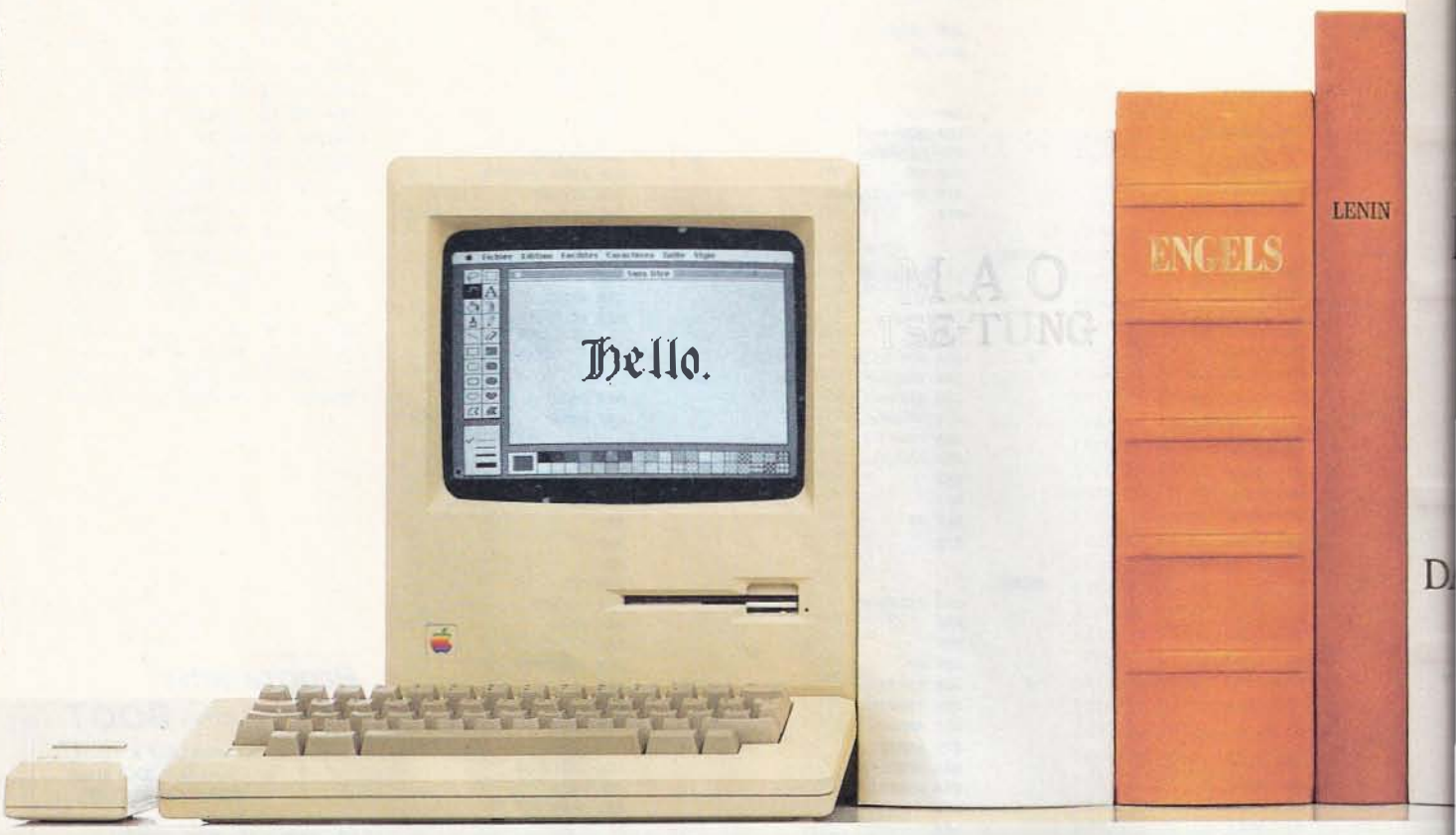
```
16A8:17 17 62 62 60 53 AA 57
16B0:21 22 03 03 00 00 02 20
16B8:01 01 00 FF 64 56 A6 56
16C0:86 85 20 4D 11 B0 4B E8
16C8:86 56 85 2D A5 19 85 2C
16D0:20 4D 11 B0 3D A0 02 C0
16D8:21 F0 E3 C8 B1 2C 24 7D
16E0:10 25 24 79 10 05 C9 FF
16E8:4C 0B 17 24 78 30 0F 29
16F0:7F 48 B1 19 29 7F 85 2E
16F8:68 C5 2E 4C 09 17 24 78
1700:10 05 C9 46 4C 09 17 D1
1708:19 F0 CC 90 B1 20 1D 17
1710:F0 AC E6 84 A5 84 C5 85
1718:D0 A2 64 85 60 A5 85 0A
1720:AA A0 02 BD 00 81 85 57
1728:BD 02 81 9D 00 81 A5 57
1730:9D 02 81 E8 88 D0 EC 60
1738:A5 85 0A AA A0 02 BD 00
1740:B1 85 57 BD FE 80 9D 00
1748:81 A5 57 9D FE 80 E8 88
1750:D0 EC 60 C6 78 D0 09 C6
1758:79 24 17 10 03 20 29 16
1760:C6 7D 64 84 20 BC 16 24
1768:79 10 10 A5 7F 85 80 0A
1770:A8 A9 00 85 81 99 00 81
1778:99 01 81 64 79 64 78 64
1780:7D 20 89 17 20 1C 14 B0
1788:FB 20 9A 0D 03 02 5E 20
1790:70 75 72 67 65 20 64 65
1798:6C 65 74 65 64 20 FD 1D
17A0:20 FC 00 A9 FF 85 60 4C
17A8:FA 11 20 89 17 10 03 20
17B0:9D 12 64 2F 20 1C 14 24
17B8:2F 30 04 C9 8D D0 F0 A5
17C0:86 49 FF 85 86 20 91 0C
17C8:03 19 FD 09 20 00 20 31
17D0:15 B0 DC 01 60 01 00 00
17D8:00 11 00 00 80 00 00 01
17E0:00 00 60 01 00 01 EF D8
17E8:00 00 00 A0 00 C8 01 00
17F0:00 00 00 00 00 00 00 00
17F8:00 00 00 00 00 00 00 00
```

Programme BOOT

Pour lancer CATS & MOUSE, faites RUN BOOT et non BRUN CATS & MOUSE...

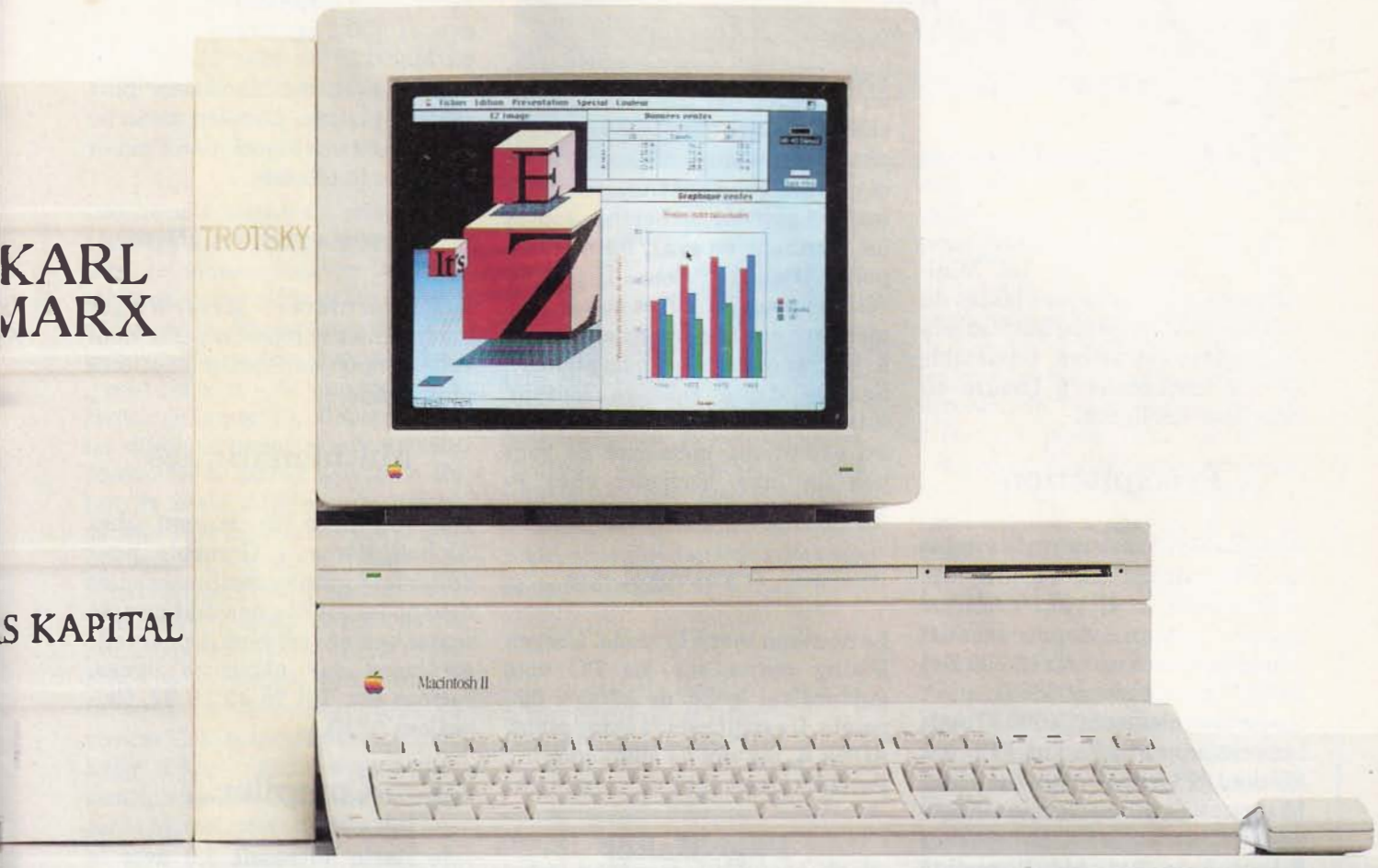
```
10 HOME : IF PEEK (49175)
   > 127 THEN PRINT "80
   COLONNES EXIGEE": NEW
20 D$ = CHR$ (13) + CHR$ (4):
   PRINT D$"PR£3"
40 POKE 1010,0: POKE 1011,3:
   POKE 1012,166
50 POKE 768,32: POKE 769,234:
   POKE 770,3
60 POKE 771,32: POKE 772,58:
   POKE 773,252
70 POKE 774,76: POKE 775,102:
   POKE 776,213
190 PRINT D$"BRUNCATS & MOUSE"
```

Il était temps qu'un révolutionnaire...



C.L.M. & BBDO. Apple, le logo Apple, Macintosh et MacPaint sont des marques déposées de Apple Computer Inc. Cricket Draw et Cricket Graph sont distribués par Alpha Systèmes.

soit aussi capable d'évoluer.



Apple

En vitrine

HyperCard

La distribution d'HyperCard reste floue, en tous cas un peu retardée, de novembre, la promesse passe à décembre pour livraison avec tous les nouveaux Mac. Les développeurs restent les privilégiés de ce programme gourmant en mémoire : moins de 1 000 F pour 360Ko de soft, des centaines de Ko de 'stackware' et la documentation. Cette documentation est en revanche à la disposition de tous chez bien des revendeurs...

3614

Le Club Apple se positionne un peu en retrait par rapport à CalvaCom mais a créé son serveur en 3614, code APPLE. 3614 à 22,20 F/heure, voilà de quoi attirer les Minitelistes applemaniaques lassés du coûteux 3615. Aujourd'hui, seule la messagerie est active, l'ensemble devant fonctionner à l'heure où vous lisez ces lignes.

Précipitation

Avec Radius Accelerator, la vitesse du MacintoshPlus ou SE est multipliée par 4, gain accentué avec l'option coprocesseur. L'accélération a un prix : 9 900 F et 4 000 F le processeur 68881...
Coûteux également (20 000 F) mais impressionnant : Radius FPD qui affiche 1024 x 864 points sur écran 15 pouces. Se connecte dans le connecteur interne du SE. Le tout chez P-Ingénierie 224, bld Raspail à Paris.

Vu...

...à Apple Expo (ou plutôt Mac-Expo) un Michel Polac effrayé par la micro : la fabuleuse démonstration de VidéoWorks y serait-elle pour quelque chose ? Plusieurs mégas pour en mettre 'plein la vue et les oreilles', mission accomplie.

Recherche attributs

Pour composer et mettre en page Quark Xpress – en français – fait un pas de plus : pour Mac Plus, SE ou II et toutes imprimantes ou photocomposeuses PostScript ou LaserWriter.
Prévu pour l'importation d'images numérisées au format TIFF, même en couleurs (on parle d'une prochaine version fonctionnant avec photocomposeuse couleur). Voilà ce qui manque à nos traitements de textes : pouvoir rechercher toutes les mentions en gras, les mots en police Palatino, etc. Problème résolu : pour 7500 F ce soft le permet ainsi que la gestion des césures à la française avec exceptions... Semble d'une relative lenteur (considérant le matériel) à l'usage, les extensions mémoires ne sont pas un luxe. Toujours chez P-Ingénierie.

Wysiwyg

Le nouveau mot à la mode. L'écran Viking connu sur les PC dote aujourd'hui le SE de 1280 x 960 points. Il est dit rapide et reposant. ACEE 4, bis rue de châteaudun à Paris.

PageMaker

Aldus Corporation annonce sa version 3.0 sur Mac et PC. Il devient possible de faire des textes importants (128 pages au lieu de 16), d'"habiller" des images, de gérer la couleur, mais toujours pas de recherche... ISE-Cégos, Tél (1) 46 84 49 84.

Ouverture

LSD Développement à Lille propose un accessoire de bureau pour les utilisateurs de LSD-Compta. Il permet la récupération de fichiers TEXT : pratique pour préparer des écritures en traitement de textes, récupérer des données de tableurs etc. Chez les concessionnaires Apple. Tiens ? LSD Compta version 2.12 permet de revenir en arrière dans la comptabilité, est-ce bien raisonnable ? Tél : 20 85 04 37

Mac contre Mac

Une occasion de faire évoluer le matériel. Voici la méthode : rendre visite à son concessionnaire Apple avec 11 750 F et un Lisa ou MacXL ou Mac 128 ou Mac 512. On en ressort avec un Macintosh Plus couleur platine. L'ancien matériel est donné à une association d'aide à l'enfance inadaptée.

Laser encore Plus

Les dernières LaserWriter américaines comportent une nouvelle version de PostScript encore plus rapide.

Mathématiques

Un accessoire de bureau chez AlphaSystème à Grenoble pour éditer les formules mathématiques devant lesquelles nos éditeurs de textes se trouvent bien dépourvus : accolades sur plusieurs lignes, racines etc. Tél 76 43 19 97, CalvaCom AS13.

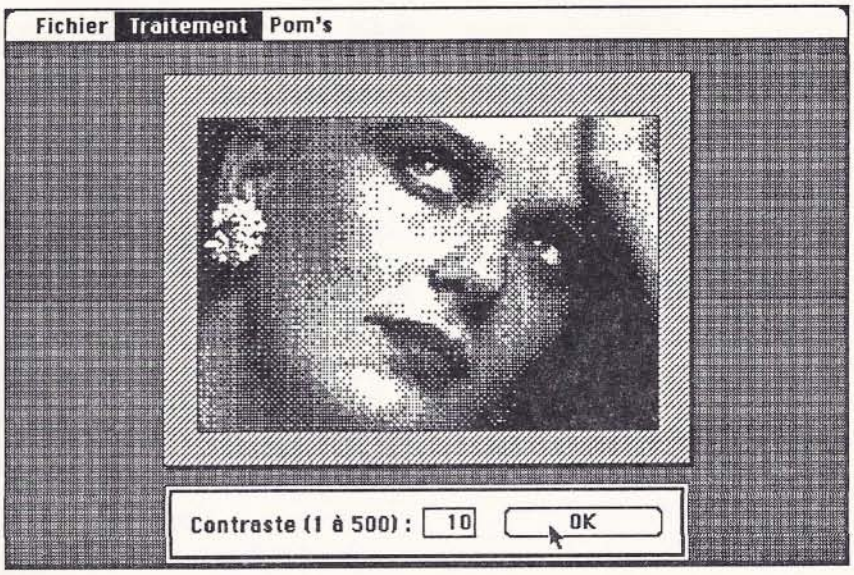
Compiler...

...le Basic Microsoft 3.0 avec le compilateur du même nom. Les résultats sont plaisants par la rapidité (ce n'est tout de même ni Pascal ni l'assembleur), par la compatibilité HFS, par les nouvelles instructions Else, EndIf, ElseIf, Case... Difficultés avec les programmes utilisant des routines assembleur. Revendeurs Apple.

Graphisme & Minitel

Jean-Luc Bazanegue

Paint → Minitel est un programme de transformation d'images MacPaint ou FullPaint en images semi-graphiques Vidéotex. Il fonctionne sur tous les Macintosh disposant d'au moins 512Ko de mémoire vive.



Directement inspiré du programme pour Apple // de Patrice Neveu publié dans ce numéro, Paint → Minitel, utilise la même méthode de calcul ; afin de ne pas doubler les explications techniques, nous vous proposons donc – pour celles-ci – de vous reporter à l'article de la page 11, où sont exposés les différents points à retenir pour l'écriture de ce genre d'application. En fait, la seule différence technique notable entre les versions Apple // et Macintosh porte sur le nombre de lignes formant l'image 'source' : 192 pour l'Apple // contre 240 (comme le Minitel) pour le Macintosh. En ce qui concerne le résultat, vous avez un exemple sur la couverture de ce numéro 33 de ce que devient la célèbre Lady d'Apple après traitement ; compte tenu des contraintes imposées par le Minitel, ce n'est pas si mal !

La liaison Macintosh/Minitel se fera, comme à l'accoutumée, par le câble de liaison que vous possédez certainement déjà. Sinon, vous pouvez le commander directement à la revue. Le raccordement se fait sur l'interface série 'communication'.

Utiliser Paint → Minitel

Si la base théorique est identique à celle de la version Apple //, il n'en va bien sûr pas de même pour la programmation et l'utilisation. Le 'listing' du programme étant suffisamment clair pour être compris aisément par nos lecteurs amateurs de Basic, nous allons tout de suite passer au mode d'emploi.

Chargement d'une image Mac/FullPaint

Le chargement d'une image 'source' est obtenu grâce à l'article *Chargement image Paint...* du menu *Fichier*, lequel provoque l'affichage de la fenêtre de sélection standard, qui elle-même ne laissera apparaître que les documents créés à l'aide de MacPaint ou FullPaint. La partie utile de l'image chargée, décodée et affichée correspond à la partie supérieure gauche du document Mac/FullPaint (240 lignes de 320 points).

Inversion de l'image 'source'

En fonction de la densité – ou plus simplement de la nature du contenu – de l'image, il peut être utile d'en inverser les valeurs ; pour cela, vous disposez de l'article *Inversion de l'image* placé dans le menu *Traitement*. Pour retrouver une image conforme au document original, il suffit d'utiliser à nouveau le même article.

Conversion de l'image

Il convient d'employer l'article *Conversion Paint/Minitel* du menu *Traitement*. Une fenêtre s'affichera alors, vous proposant un traitement en noir et blanc ou avec des nuances de gris. Avec le traitement en noir et blanc, seules ces deux valeurs extrêmes seront utilisées sur l'écran du Minitel ; cette méthode se révèle être souvent efficace avec les images faiblement contrastées. Lors de l'emploi des nuances, une seconde fenêtre vous propose le choix du contraste, de 1 à 500. Par défaut, le contraste est de 10 pour les images 'normales', et 100 pour les images inversées.

Pendant le traitement proprement dit, un rectangle visualisant le caractère concerné ce déplace rapidement sur l'écran du Macintosh, alors que le résultat s'affiche simultanément sur l'écran de votre Minitel. Bien que la présence de ce dernier soit pratique – puisqu'elle autorise un contrôle immédiat du résultat – elle n'est absolument pas indispensable.

Les opérations peuvent être interrompues en maintenant enfoncées les touches Commande et '.'.

Sauvegarde de l'image Vidéotex

Après traitement, l'image Vidéotex est en mémoire et, si nous voulons la conserver pour une utilisation future, il faut l'enregistrer avec l'article *Sauvegarde image Minitel...* du menu *Fichier*. Une fenêtre standard permet de nommer le fichier qui reçoit l'œuvre. Pour une exploitation des images issues de Paint → Minitel avec un autre logiciel (serveur par exemple), signalons que le type donné aux fichiers générés est *ImMi*, pour Image Minitel (l'imagination, quand même...).

Consultation d'une image Vidéotex

Il est possible de charger rapidement et d'envoyer vers le Minitel des images Vidéotex en utilisant l'article *Chargement & envoi d'une image Minitel...* du menu *Fichier*. Dans ce cas – et logiquement – seuls les fichiers de type *ImMi* peuvent être sélectionnés.

Exemple d'utilisation des images créées

Ces images seront évidemment utilisables par votre serveur actuel ou futur mais, en attendant, il est déjà possible de l'envoyer sur le Minitel d'un correspondant avec le petit programme – rustique – suivant :

```
DEFINT A-Y
Escape$=CHR$(27)
PRO1$=Escape$+CHR$(57)
SOH$=CHR$(1)
EOT$=CHR$(4)
ENQROM$=CHR$(&H7B)
Deconnexion$=CHR$(103)
Connexion$=CHR$(104)
RET1$=CHR$(111)
MENU 1,0,1,"Fichier"
MENU 1,1,1,"Ouvrir..."
MENU 1,2,1,"-"
MENU 1,3,1,"Quitter"
WINDOW CLOSE 1
ON MENU GOSUB EventMenu
MENU ON
ON BREAK GOSUB rien
BREAK ON
Boucle:
GOTO Boucle
EventMenu:
MENU OFF
Nmenu=MENU(0)
Narticle=MENU(1)
ON Nmenu GOSUB mFichier,rien
MENU
MENU ON
RETURN

mFichier:
ON Narticle GOSUB aChargeMinitel, ∫
    rien,Quitter
RETURN
aChargeMinitel:
ON ERROR GOTO ErreurChargeMinitel
FichierMinitel$=FILES$(1,"ImMi")
IF FichierMinitel$="" THEN
RETURN
ENDIF
OPEN FichierMinitel$ FOR INPUT ∫
    AS 1 LEN=512
ChaineImage$=INPUT$(LOF(1),1)
CLOSE 1
OPEN "COM1:1200" FOR OUTPUT ∫
    AS 1 LEN=512
OPEN "COM1:1200" FOR INPUT AS 2
PRINT#1,PRO1$ RET1$ PRO1$ ∫
    Connexion$;
tempo#=TIMER
WHILE TIMER<>tempo#+6
WEND
Envois=0
Boucle2:
tempo#=TIMER
PRINT#1,PRO1$ ENQROM$;
WHILE TIMER<tempo#+1
IF LOC(2)>=3 THEN GOSUB Test
WEND
Envois=Envois+1
IF Envois<20 THEN Boucle2
PRINT#1,PRO1$ Deconnexion$;
CLOSE 1
CLOSE 2
RETURN
ErreurChargeMinitel:
RESUME aChargeMinitel
Quitter:
SYSTEM
rien:
RETURN
Test:
GOSUB LitCar
```

Traitement

Inversion de l'image
Conversion Paint/Minitel

```
IF C$<>SOH$ THEN RETURN
GOSUB LitCar
GOSUB LitCar
GOSUB LitCar
GOSUB LitCar
IF C$<>EOT$ THEN RETURN
PRINT#1,ChaineImage$;
Envois=19
RETURN
LitCar:
C$=INPUT$(1,2)
RETURN
```

Lorsque qu'une image est sélectionnée, votre correspondant dispose de 20 secondes – à partir du moment où il entend la porteuse (signal aigu) – pour appuyer sur la touche *Connexion* de son Minitel. Il commencera alors à recevoir l'image dans la seconde qui suit la connexion. Attention : le Minitel émetteur doit être 'retournable' (Type 1B).

Sur la disquette d'accompagnement, l'application correspondante s'appelle *Envoi Vidéotex*.

Programme Basic

L'application Paint → Minitel, que l'on peut trouver sur la disquette d'accompagnement du numéro 33, est écrite avec la version 3.0 du Basic Microsoft et compilée avec le compilateur du même nom. Si vous n'avez pas ladite disquette d'accompagnement et voulez utiliser le programme avec l'interpréteur, Paint → Minitel fonctionnera mais, bien sûr, beaucoup plus lentement. D'autre part, nous avons utilisé la possibilité de *IF... THEN... ELSE IF... ENDIF* de la version 3.0 qui n'est pas tolérée par les versions antérieures ; si vous êtes munis d'une de ces versions, il vous faudra donc faire quelques menues adaptations. Ainsi, par exemple :

```
MiseAJour:
IF DrapeauInvert THEN
PUT (0,0),TableauMAJ,PRESET
ELSE
PUT (0,0),TableauMAJ,PSET
```

```
END IF
RETURN
```

deviendra :

```
MiseAJour:
IF DrapeauInvert THEN PUT
   (0,0),TableauMAJ,PRESET
   ELSE PUT(0,0),TableauMAJ,
   PSET
RETURN
```

Fichier

Chargement image Paint...
Chargement & envoi image Minitel...
Sauvegarde image Minitel...

Quitter

Programme 'Paint -> Minitel'

Basic Microsoft 3.0

** le symbole '∫' indique la continuité de la ligne.*

```
'*****************************************
'* Conversion d'images Mac/FullPaint
'* en Images Semi-graphiques Vidéotex
'*****************************************
'* Initialisations
DEFINTA-Y
DATA &H4E56,&H0000,&H3F2E,&H0008
DATA &HA936,&HA937,&H4E5E,&H4E75
DATA &H4E56,&H0000,&H42A7,&H3F2E
DATA &h0008,&hA9B9,&h2057,&h2E90
DATA &HA851,&H4E5E,&H4E75
DATA 0,4,1,5,2,6,3,7
'Haut gauche
DATA 0,3,0,2
'haut droit
DATA 4,7,0,2
'milieu gauche
DATA 0,3,3,6
'milieu droit
DATA 4,7,3,6
'bas gauche
DATA 0,3,7,9
'bas droit
DATA 4,7,7,9
DIM TableauEntiers(38),TableauMAJ(4801)
DIM TableauPoints(39,23,5),TableauDensite(7)
DIM Gris(3),Rectangle(3),T(5),TableauHexa(14)
DIM DepartX(5),ArriveeX(5),DepartY(5),ArriveeY(5)
DIM SupMenus(7),Curseurs(10)
NombreLignes=240
NombreMots=20
TableauEntiers(1)=320
TableauEntiers(2)=1
TableauMAJ(0)=320
TableauMAJ(1)=240
EOR=10
top=0
left=1
bottom=2
right=3
MenuPomme=1
Montre=4
Chicago=0
FormFeed$=CHR$(12)
SemiGraph$=CHR$(14)
Texte$=CHR$(15)
Escape$=CHR$(27)
FORi=0TO 7
   READ SupMenus(i)
NEXT
FOR i=0 TO 10
   READ Curseurs(i)
NEXT
FOR i=0 TO 7
   READ TableauDensite(i)
NEXT
FOR i=0 TO 3
   Gris(i)=&HAA55
NEXT
FOR i=0 TO 5
   READ DepartX(i),ArriveeX(i),DepartY(i),∫
   ArriveeY(i)
NEXT
FOR i=0 TO 14
   TableauHexa(i)=2^i
NEXT
A!=VARPTR(SupMenus(0))
A! MenuPomme
MENU 1,0,1,"Fichier"
MENU 1,1,1,"Chargement image Paint..."
MENU 1,2,1,"Chargement & envoi image Minitel..."
MENU 1,3,0,"Sauvegarde image Minitel..."
MENU 1,4,0," "
MENU 1,5,1,"Quitter"
MENU 2,0,1,"Traitement"
MENU 2,1,0,"Inversion de l'image"
MENU 2,2,0,"Conversion Paint/Minitel"
MENU 3,0,1,"Pom's"
MENU 3,1,1,"©JLB, CP & Éditions MEV ∫
— 39.51.24.43"
WINDOW CLOSE 1
ON DIALOG GOSUB EventDialogue:DIALOG ON
ON MENU GOSUB EventMenu:MENU ON
ON BREAK GOSUB rien:BREAK ON

'*****************************************
'* Boucle principale

Boucle:
GOTO Boucle
EventDialogue:
   DIALOG OFF
   GOSUB CurseurMontre
      GOSUB MiseAJour
   DIALOG ON
   INITCURSOR
RETURN
```

```
EventMenu:
   MENU OFF:DIALOG OFF
      GOSUB CurseurMontre
      Nmenu=MENU(0)
      Narticle=MENU(1)
      ON Nmenu GOSUB mFichier,mTraitement,rien
      MENU
   MENU ON:DIALOG ON
   INITCURSOR
RETURN
mFichier:
   ON Narticle GOSUB aChargePaint,∫
   aChargeMinitel,aSauveMinitel,Arret,Quitter
RETURN
aChargePaint:
GOSUB OuverturePaint
IF NOT Annuler THEN
   GOSUB PaintBitMap
   MENU 2,1,1
   MENU 2,2,1
   MENU 1,3,0
END IF
RETURN
aChargeMinitel:
   GOSUB ChargementMinitel
RETURN
aSauveMinitel:
   GOSUB Sauvegarde
RETURN
mTraitement:
   ON Narticle GOSUB aInversion,aConversion
RETURN
aInversion:
   GOSUB Inversion
RETURN
aConversion:
   GOSUB CouleurOuNoir
   GOSUB ComptePoints
RETURN

'*****************************************
'* Ouverture d'un fichier MacPaint ou FullPaint
OuverturePaint:
ON ERROR GOTO ErreurOuverture
   FichierPaint$=FILES$(1,"PNTG")
   GOSUB CurseurMontre
   IF FichierPaint$="" THEN
      Annuler=-1
      GOSUB MiseAJour
      RETURN
   END IF
   Annuler=0
   OPEN FichierPaint$ FOR INPUT AS 1 LEN=512
   CLS
   DrapeauInvert=0
   IF NOT DrapeauFenetre THEN
   WINDOW 1,"",(96,40)-(416,280),4
   DrapeauFenetre=-1
   END IF
RETURN
ErreurOuverture:
   RESUME OuverturePaint

'*****************************************
'* Décodage et affichage de l'image Paint
PaintBitMap:
TotalPoints!=0
VariableBidon$=INPUT$(512,1)
PENPAT VARPTR(Gris(0))
FOR Indice1=1 TO NombreLignes
   MOVETO 0,Indice1-1
   LINETO 319,Indice1-1
   FOR Indice2=1 TO NombreMots
      TableauEntiers(Indice2+2)=0
   NEXT
   CompteurOctets=0
BoucleLigne:
   NombreOctets=ASC(INPUT$(1,1))
   IF NombreOctets<185 THEN
      VraiNombreOctets=NombreOctets+1
      GOTO CodeDifferents
      ELSE
         VraiNombreOctets=256-NombreOctets+1
         ValeurReport=ASC(INPUT$(1,1))
      END IF
   FOR Indice2=1 TO VraiNombreOctets
      IF (Indice2+CompteurOctets+1) MOD 2 THEN
         TableauEntiers((Indice2+CompteurOctets+1)\∫
         2+2)=TableauEntiers((Indice2+∫
         CompteurOctets+1)\2+2) OR ValeurReport
         GOTO FinIndice2
      END IF
      ValeurReportLong!=ValeurReport*256
      IF ValeurReportLong!>32767 THEN
         ValeurReportLong!=ValeurReportLong!-65536!
      END IF
      ValeurReportCourt=ValeurReportLong!
      TableauEntiers((Indice2+CompteurOctets+1)\2+∫
      2)=TableauEntiers((Indice2+CompteurOctets+∫
      1)\2+2) OR ValeurReportCourt
FinIndice2:
   NEXT
   CompteurOctets=CompteurOctets+VraiNombreOctets
   GOTO SuiteLigne
CodeDifferents:
   FOR Indice2=1 TO VraiNombreOctets
   CompteurOctets=CompteurOctets+1
   ValeurReport=ASC(INPUT$(1,1))
   IF (CompteurOctets+1) MOD 2 THEN
      TableauEntiers((CompteurOctets+1)\2+2)=∫
      TableauEntiers((CompteurOctets+1)\2+2) OR ∫
      ValeurReport
      GOTO FinIndice2Bis
   END IF
   ValeurReportLong!=ValeurReport*256
   IF ValeurReportLong!>32767 THEN
      ValeurReportLong!=ValeurReportLong!-65536!
   END IF
   ValeurReportCourt=ValeurReportLong!
   TableauEntiers((CompteurOctets+1)\2+2)=∫
   TableauEntiers((CompteurOctets+1)\2+2) OR ∫
   ValeurReportCourt
FinIndice2Bis:
   NEXT
SuiteLigne:
   IF CompteurOctets<72 THEN BoucleLigne
   FOR Indice2=1 TO NombreMots
      TableauMAJ((Indice1-1)*20+(Indice2+1))=∫
      TableauEntiers(Indice2+2)
```

```
      FOR Indice3=0 TO 14
        IF (TableauEntiers(Indice2+2) AND ∫
        TableauHexa(Indice3))=0 THEN
          TotalPoints!=TotalPoints!+1
        END IF
      NEXT
      IF TableauEntiers(Indice2+2)<0 THEN ∫
      TotalPoints!=TotalPoints!+1
    NEXT
    PUT(0,Indice1-1),TableauEntiers(1),PSET
NEXT
CLOSE 1
MI#=((TotalPoints!/5760)*27)/30
PENNORMAL
RETURN

'*****************************************
'* Choix du traitement
CouleurOuNoir:
  WINDOW 2,"",(104,300)-(408,332),2
  BUTTON 1,1,"Noir et blanc",(8,7)-(144,25)
  BUTTON 2,1,"Nuances grisées",(160,7)-(296,25)
  INITCURSOR
  WHILE DIALOG(0)<>1
  WEND
    IF DIALOG(1)=1 THEN
    DrapeauNoir=-1
    WINDOW CLOSE 2
    RETURN
      ELSE
      DrapeauNoir=0
      BUTTON CLOSE 2
      TEXTFONT Chicago
      MOVETO 8,21
      BUTTON 1,1,"OK",(200,7)-(296,25)
      PRINT "Contraste (1 à 500) :";
      IF DrapeauInvert THEN
        BaseContraste$="100"
        ELSE
          BaseContraste$="10"
      END IF
BoucleContraste:
      EDIT FIELD 1,BaseContraste$,(152,8)-∫
      (180,23),1,3
        WHILE DIALOG(0)
        WEND
      Dialogue=DIALOG(0)
      WHILE Dialogue<>6 AND Dialogue<>1
      Dialogue=DIALOG(0)
      WEND
      Contraste#=VAL(EDIT$(1))
      IF Contraste#<1 OR Contraste#>500 THEN
        BEEP
        GOTO BoucleContraste
      END IF
      GOSUB CurseurMontre
      WINDOW CLOSE 2
  END IF
RETURN

'*****************************************
'* Rafraîchissement de l'écran
MiseAJour:
  IF DrapeauInvert THEN
    PUT (0,0),TableauMAJ,PRESET
  ELSE
    PUT (0,0),TableauMAJ,PSET
  END IF
RETURN

'*****************************************'
* Transformation en image Semi-graphique
ComptePoints:
OPEN "COM1:1200" FOR OUTPUT AS 1 LEN=512
PRINT#1,FormFeed$ SemiGraph$;
PENPAT VARPTR(Gris(0))
PENMODE EOR
TEXTFONT Chicago
CouleurFondBis=0
CouleurCaractereBis=0
ChaineImage$=""
DrapeauArret=0
GOSUB FenetreArret
GOSUB CurseurMontre
MENU 1,3,1
ON BREAK GOSUB Arret
FOR Vertical=0 TO 23
  HIDECURSOR
  FOR Horizontal=0 TO 39
  Rectangle(top)=Vertical*10-1
  Rectangle(left)=Horizontal*8-1
  Rectangle(bottom)=Vertical*10+11
  Rectangle(right)=Horizontal*8+9
  FRAMERECT(VARPTR(Rectangle(top)))
  S=0
    FOR Bloc=0 TO 5
      TableauPoints(Horizontal,Vertical,Bloc)=0
      FOR PointX=DepartX(Bloc) TO ArriveeX(Bloc)
        FOR PointY=DepartY(Bloc) TO ∫
        ArriveeY(Bloc)
          TableauPoints(Horizontal,Vertical,∫
          Bloc)=TableauPoints(Horizontal,∫
          Vertical,Bloc)+ABS(POINT(Horizontal*∫
          8+PointX,Vertical*10+PointY)=30)
        NEXT
      NEXT
      IF Bloc=2 OR Bloc=3 THEN
      TableauPoints(Horizontal,Vertical,Bloc)=∫
      (TableauPoints(Horizontal,Vertical,Bloc)*3)\4
      END IF
      T(Bloc)=TableauPoints(Horizontal,Vertical,∫
      Bloc)
      S=S+T(Bloc)
    NEXT
    IF DrapeauNoir THEN
      CouleurFond=7
      CouleurCaractere=0
      GOTO Suite0
    END IF
    Ecart#=((T(0)^2+T(1)^2+T(2)^2+T(3)^2+∫
    T(4)^2+T(5)^2)-(S^2)/6)/5
    Ecart#=(Ecart#*Contraste#)/10
    CouleurCaractere=INT((S/6-Ecart#/2)+.5)
    CouleurFond=INT((S/6+Ecart#/2)+.5)
    IF CouleurCaractere>7 THEN ∫
    CouleurCaractere=7:GOTO Suite1
    IF CouleurCaractere<0 THEN ∫
    CouleurCaractere=0
Suite1:
    IF CouleurFond>7 THEN CouleurFond=7:GOTO
Suite0
```

```
    IF CouleurFond<0 THEN CouleurFond=0
Suite0:
    Caractere=32+(ABS(T(0)>MI#))+2*(ABS(T(1)>∫
    MI#))+4*(ABS(T(2)>MI#))+8*(ABS(T(3)>MI#∫
    ))+16*(ABS(T(4)>MI#))+64*(ABS(T(5)>MI#))
    CouleurFond=TableauDensite(CouleurFond)
    CouleurCaractere=TableauDensite∫
    (CouleurCaractere)
    IF CouleurFond<>CouleurFondBis THEN
      PRINT#1,Escape$ CHR$(64+CouleurFond);
      ChaineImage$=ChaineImage$+Escape$+∫
      CHR$(64+CouleurFond)
      CouleurFondBis=CouleurFond
    END IF
    IF CouleurCaractere<>CouleurCaractereBis ∫
    THEN
      PRINT#1,Escape$ CHR$(80+∫
      CouleurCaractere);
      ChaineImage$=ChaineImage$+Escape$+∫
      CHR$(80+CouleurCaractere)
      CouleurCaractereBis=CouleurCaractere
    END IF
    PRINT#1,CHR$(Caractere);
    ChaineImage$=ChaineImage$+CHR$(Caractere)
  FRAMERECT(VARPTR(Rectangle(top)))
  IF DrapeauArret THEN
    GOSUB MiseAJour
    Vertical=23
    Horizontal=39
    MENU 1,3,0
  END IF
  NEXT
NEXT
PRINT#1,Texte$;
ChaineImage$=FormFeed$+SemiGraph$+∫
ChaineImage$+Texte$
WINDOW CLOSE 2
SHOWCURSOR
CLOSE 1
ON BREAK GOSUB rien
RETURN

'*****************************************'
'* Enregistrement de l'image semi-graphique
Sauvegarde:
ON ERROR GOTO ErreurSauvegarde
  FichierMinitel$=FILES$(0,"Enregistrer l'image ∫
  sous :")
  GOSUB CurseurMontre
  GOSUB MiseAJour
  IF FichierMinitel$="" THEN
    Annuler=-1
    RETURN
  END IF
  Annuler=0
  OPEN FichierMinitel$ FOR OUTPUT AS 1 LEN=∫
  512
  PRINT#1,ChaineImage$;
  CLOSE 1
  NAME FichierMinitel$ AS FichierMinitel$,∫
  "ImMi"
RETURN
ErreurSauvegarde:
  RESUME Sauvegarde

'*****************************************
```

```
'* Chargement d'une image Semi-graphique
'* et envoi vers le Minitel
ChargementMinitel:
ON ERROR GOTO ErreurChargeMinitel
  FichierMinitel$=FILES$(1,"ImMi")
  GOSUB CurseurMontre
  GOSUB MiseAJour
  IF FichierMinitel$="" THEN
    Annuler=-1
    RETURN
  END IF
  Annuler=0
  OPEN FichierMinitel$ FOR INPUT AS 1 LEN=512
  ChaineImage$=INPUT$(LOF(1),1)
  CLOSE 1
  OPEN "COM1:1200" FOR OUTPUT AS 1 LEN=512
  PRINT#1,ChaineImage$;
  CLOSE 1
RETURN
ErreurChargeMinitel:
  RESUME ChargementMinitel

'*****************************************
'* Inversion (noir/blanc) de l'image Paint
Inversion:
  DrapeauInvert=NOT DrapeauInvert
  Rectangle(top)=0
  Rectangle(left)=0
  Rectangle(bottom)=240
  Rectangle(right)=320
  INVERTRECT(VARPTR(Rectangle(top)))
RETURN

'*****************************************
'* Sous-programme divers
Arret:
  DrapeauArret=-1
RETURN
Quitter:
SYSTEM
CurseurMontre:
  A!=VARPTR(Curseurs(0))
  A! Montre
RETURN
FenetreArret:
  WINDOW 2,"",(104,300)-(408,332),2
  TEXTFONT Chicago
  MOVETO 32,21
  PRINT "Pour annuler : " CHR$(17)" - majuscule ∫
  - '.'...";
  WINDOW 1
RETURN
rien:
RETURN
'*****************************************
```


Vous cherchez

désespéremment un ordinateur, vous vendez une souris, vous avez créé un club ?

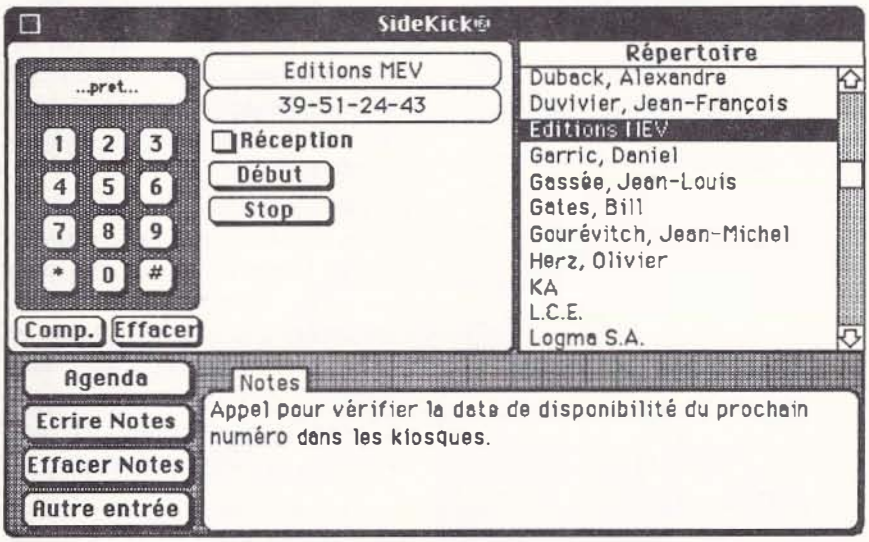
Rendez-vous page 29...

A l'essai : SideKick version 2

Hervé Thiriez

Livrée avec une documentation claire et bien ficelée de 240 pages et deux disquettes, la deuxième version de SideKick regroupe les accessoires de bureau suivants, la taille mémoire en octets étant indiquée entre parenthèses :

- Agenda (2468) : calendrier mensuel avec des possibilités d'entrer des notes pour chaque jour et d'afficher un calendrier hebdomadaire ;
- Bloc-Notes+ (10 978) : un mini-traitement de texte (inspiration MockWrite) qui génère des documents en format ASCII et comporte des fonctions d'édition et de recherche ;
- Calculatrice+ (6 020) : une calculatrice professionnelle multi-fonctions avec bande-papier témoin imprimable ;
- Composeur (2 447) : un composeur téléphonique ;
- Heure (1 060) : horloge analogique (inspiration Big Ben) avec aiguille des secondes ;
- Idées+ (113 057) : un processeur d'idées (inspiration Think Tank) ;
- MacPlan (40 139) : un tableur avec possibilités graphiques ;
- Mémo-Listes (2433) : quatre listes prédéfinies servant d'aide-mémoire ;
- ReadyPrinter (7 013) : un *spooler* qui imprime les fichiers ASCII en vous laissant l'accès au clavier et à la souris ;
- Terminal (14 224) : programme de communication pour modems de type Hayes 300, 1200 et 2400 bauds. Intégré avec Bloc-Notes+ pour l'émission, la réception et l'édition des données.



En plus de tout cela, l'application **SideKick** offre une version plus riche et plus élaborée des programmes de renseignements personnels, de composition téléphonique et de gestion du temps que ceux fournis par les accessoires.

Enfin, les applications *Idées+*, *Configurer Terminal*, *Copie Répertoire*, *Editeur Mémo-Listes* et *PrintManager* vous rendent un certain nombre de services complémentaires.

Nous pouvons regretter qu'il soit nécessaire de loger dans le même lecteur que le fichier système les fichiers de données de SideKick : cela peut vite devenir encombrant pour ceux qui n'utilisent pas de disque dur, surtout si on tient compte de la place prise par les accessoires de SideKick.

SideKick est avant tout un outil de communication. En effet, dès l'appel du programme, une fenêtre apparaît (voir copie d'écran), où l'on voit que l'aspect communication domine, ne serait-ce que géographiquement.

Un bon exemple de convivialité : un clic sur un nom dans la zone *Répertoire* affiche les données relatives à ce nom, un second clic réaffiche la liste des noms. Un double clic sur un nom ou une fiche fait apparaître la totalité de la fiche en mode *Edition*. Mon seul regret à ce niveau est que la barre des menus disparaît alors, ce qui interdit de couper/coller si, par exemple, l'on a entré des données au mauvais endroit et que l'on cherche à les déplacer.

L'Agenda

L'Agenda est bien pratique, surtout si vous avez une activité sédentaire et gardez votre Mac allumé tout le temps. Il semble avoir un problème avec les systèmes français (au moins en version 4.0) : en effet, dans la présentation hebdomadaire, il n'affiche pas le nombre des unités pour la date du jour, mais seulement le nombre des dizaines... Il est regrettable aussi que la présentation hebdomadaire ne soit possible que pour trois semaines (semaine passée, présente et future) : on est encore loin de la flexibilité offerte par ce merveilleux outil rétro

qu'est l'agenda papier !

Bloc-Notes

Cela ressemble parfaitement à MockWrite, avec la possibilité de rechercher (mais pas de remplacer).

Idées+

C'est en fait un Think Tank plus qu'un More. Toute rubrique peut être ouverte et développée en un texte complet : très agréable. Au chapitre des imperfections : le remplacement avec intervention manuelle semble ne pas marcher du tout (pire, il ajoute purement et simplement le texte de remplacement là où se trouve le curseur), sauf s'il est systématique ; et, dans ce cas, il va seulement du curseur à la fin du texte sans faire un tour complet de celui-ci.

Si MacWrite 4.6 présente encore des améliorations, il a un côté rassurant par la persistance d'au moins un bug, pas de dépaysement !

La méthode pour obtenir une bombe demeure la suivante :

Dans la fenêtre de recherche, saisir un mot, effacer un caractère à l'aide de BackSpace, puis tenter de le reposer par Commande-BackSpace...

Autre problème, mais la version 4.6 n'est pas responsable, Orthogiciel 2 semble un peu perdu lors de la réécriture des fichiers : normal, il a été conçu avec la version 4.5.

MacPlan

C'est uniquement en épluchant la documentation que j'ai découvert l'existence de fonctions dans MacPlan. Celles-ci ne sont pas visibles facilement (pas de *Coller une fonction*, de liste des fonctions à l'écran ou d'écrans d'aide). Autre désagrément : quand on ferme MacPlan en ayant oublié de sauvegarder le tableau réalisé, celui-ci est perdu : il n'y a en effet pas de message d'avertissement demandant confirmation de la fermeture sans sauvegarde, que ce soit après création ou après modification.

La possibilité d'obtenir rapidement un graphe est bien agréable, mais le mode de titrage est très lourd : on ne peut pas, comme dans Excel, sélectionner à la fois les légendes et les valeurs ; plusieurs étapes sont nécessaires pour ajouter à un graphe un titre, une légende, et des indications sur l'axe des abscisses.

Mémo-Listes

Cet accessoire gère quatre feuillets, que l'on fait défiler comme ceux du Calepin : *Alarmes*, *Cartes de crédit*, *Choses à faire* et *Dépenses*. Il me manque, dans *Choses à faire*, soit la possibilité de facilement modifier l'ordre des choses à faire (dont l'urgence relative varie dans le temps), soit celle d'attribuer à chaque chose à faire un numéro d'urgence en fonction de quoi tout se classe. Je continue donc à utiliser Excel à cet effet, MacPlan étant incapable de prendre en charge ce classement puisqu'il ne comporte pas de facilité de tri.

Fiabilité

Je me suis trouvé plusieurs fois en panne, soit avec des bombes, soit – lors de l'appel d'un accessoire – avec un curseur totalement bloqué : dans les deux cas de figure, bien entendu, il a fallu redémarrer. Il semble que ces ennuis surviennent quand on utilise le Switcher : c'est bien malheureux car une application comme SideKick est normalement faite pour rester disponible à tout moment en arrière-plan grâce à un outil tel que le Switcher. Ainsi, pour la rédaction de l'article, j'avais MacWrite et SideKick en parallèle grâce au Switcher ; et j'ai rapidement appris à effectuer une sauvegarde avant chaque nouvel essai d'un accessoire ou de l'application SideKick.

Conclusion

En conclusion, SideKick 2.0 est un produit sérieux, bien documenté, et avec des possibilités attrayantes, mais comportant quelques problèmes de finition. Ce que je lui reproche le plus, Borland n'y pouvant vraiment rien, c'est de ne pas pouvoir l'utiliser sur un Mac portable et fonctionnant sur batteries, qui m'accompagnerait dans tous mes déplacements. En effet, un agenda – par exemple – doit être toujours accessible. C'est pourquoi je vois bien SideKick utilisé par les personnes travaillant à poste fixe : un excellent outil pour sédentaire. Un de mes amis, ayant un Mac au bureau et un à la maison, me dit qu'il s'en sert souvent à condition de toujours garder sur lui la disquette avec les fichiers SideKick : certes, c'est une solution possible, mais qui n'est pas d'une grande simplicité !

Au chapitre de la francisation du logiciel, on peut regretter que les raccourcis n'aient pas été adaptés : utiliser F (Find) pour *Rechercher* et N (Next) pour *Suivant* n'est pas vraiment convivial. Il semble aussi que le logiciel original n'ait pas bien respecté les normes de programmation indiquées par Apple aux développeurs, ce qui expliquerait le bug de l'agenda, où le chiffre des unités manque dans la présentation hebdomadaire.

En 'bonus' sur la disquette Pom's Mac 33, nous avons placé un 'Coucou' somptueux écrit par notre ami et collaborateur Alain Bohec. Il affiche, outre l'heure, la date et la fête du jour. Si vous démarrez votre Macintosh avec la disquette en question, le

coucou sera lancé automatiquement et, dans ce cas, il vous faut savoir qu'il est nécessaire d'appuyer sur la touche 'F' (comme Fin ou Finder) pour en sortir. Signalons aussi qu'une pression sur la souris fait sortir le coucou !

Recherche d'applications

Georges Zwingelstein

Le système de dossiers et fichiers hiérarchisé du Macintosh Plus est nettement plus pratique que l'ancien système MFS. De même, le disque dur est intéressant pour le nombre important d'applications et de documents qu'il peut contenir. Le problème est de retrouver l'application dont on a besoin, en particulier lorsqu'on a l'habitude de travailler avec les dossiers, sous-dossiers, sous-sous-dossiers..., ou lorsque plusieurs personnes utilisent le même disque dur, chacune ayant ses propres dossiers.

Deux solutions s'offrent alors :

- dupliquer l'application dans chaque dossier où elle s'avère nécessaire, ce qui prend inutilement de la place ;
- faire la recherche manuelle de l'application, en prenant soin de revenir dans le bon dossier lors de l'enregistrement.

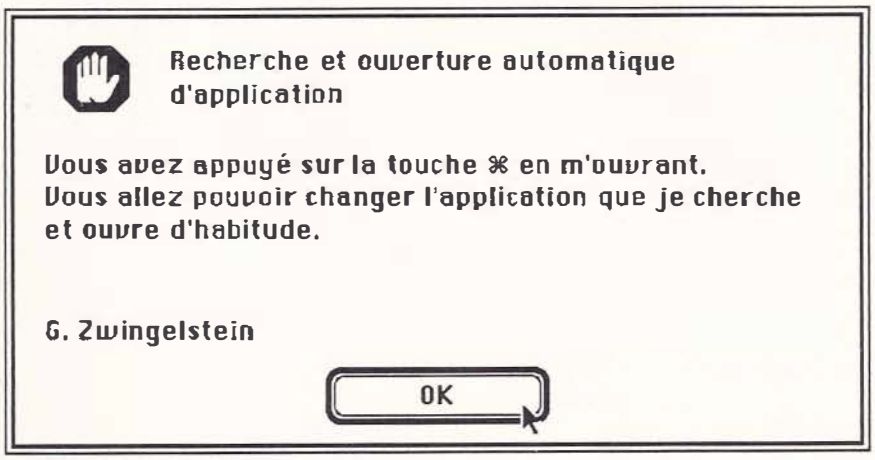
Pom's vous propose une troisième solution qui consiste en une petite application – 'Appl Search' – de quelques kilooctets seulement, donc dupliquable sans perte importante de place.

La première fois, elle vous propose de choisir une application, et par la suite elle cherche cette application sur un des disques connus par le Mac (en ligne), y compris les disques en HFS, et la lance si elle l'a trouvée.

Vous pouvez renommer 'Appl Search' à votre guise et, en particulier, lui donner le même nom que l'application cherchée puisque :

- la recherche se fait à la fois sur le nom et sur le créateur (signature comme, par exemple, 'Ta2n' pour InterPom's) ;
- il n'est possible de choisir que des applications ayant un créateur (ce qui est le cas de la grande majorité des applications intéressantes) ;
- 'Appl Search' n'a pas de créateur.

Il n'y a donc pas de risque de bouclage avec une autre application du type 'Appl Search' se trouvant dans un autre disque ou dossier.



Si vous désirez changer d'application par la suite, il suffit d'appuyer sur la touche commande au moment du lancement de 'Appl Search'. Il vous sera proposé le choix d'une nouvelle application. Si vous annulez 'Appl Search' redeviendra comme lors de sa première utilisation.

Afin que la recherche soit la plus rapide possible, il est conseillé de mettre les applications dans un des premiers dossiers dans l'ordre alphabétique. En effet, la recherche dans les dossiers se fait suivant cet ordre.

Note : il existe des applications qui modifient certaines procédures du file manager. *Celles-ci font alors la recherche d'un fichier dans tous les dossiers et sous-dossiers des disques formatés en HFS, et permettent alors à certaines applications non compatibles avec ce système de fonctionner quand même. Nous connaissons en particulier 'HFSOpen' dont les modifications de recherche automatique sont commandées par la touche option. Avec les modifications de cette application, 'Appl Search' peut donner une erreur système si la touche option est enfoncée, car la recherche dans le HFS se fait même avec la fonction PBHGetFInfo, ce qui n'est pas normal puisqu'une application utilisant cette fonction est sensée fonctionner en HFS. Donc, ne pas appuyer sur la touche option lors du lancement de 'Appl Search', ou bien appuyer jusqu'au lancement effectif de l'application recherchée.*

Fichier 'Appl Search.Link'

```
; le résultat sera baptisé 'Appl Search
; (par défaut)

; commence par le fichier .Rel
Appl Search

; ajoute maintenant les ressources
/Include Appl Search.rsrc

; terminé.
/End
```

Fichier 'Appl Search.Job'

Asm	Appl Search.Asm	Exec	Edit
RMaker	Appl Search.R	Exec	Edit
Link	Appl Search.Link	Finder	Edit

Fichier 'Appl Search.R'

```
* le symbole 'ʃ' indique la
continuité de la ligne

*
* resource pour l'application Appl Search
*

Appl Search.rsrc

Type ALRT
 ,128
50 50 250 450
128
4444

* DITL de la fenetre d'aterte
Type DITL
 ,128
4

Button
170 150 190 250
OK

StaticText Disabled
10 7050 390
Recherche et ouverture automatique
d'application

StaticText Disabled
60 10 130 390
^0^1^2^3

StaticText Disabled
140 10 160 390
G. Zwingelstein

Type DLOG
 ,129
Recherche
50 50 250 450
visible NoGoAway
0
0
129

* DITL du dialog de recherche
Type DITL
 ,129
4

Button
170 150 190 250
Annuler

StaticText Disabled
10 10 50 390
Je cherche l'application ^0.

StaticText Disabled
60 10 130 390
Pour arrêter la recherche et revenir au ʃ
bureau, cliquer sur ++
le bouton 'annuler'^1

StaticText Disabled
140 10 160 390
G. Zwingelstein

; resource non purgeable
Type GNRL
type et créateur ,256 (0)
.S
????????

; resource non purgeable
Type STR
Nom ,256 (0)


 ,257 (32)
Je ne sais pas encore quelle application jeʃ
dois chercher et ouvrir.\0D++
Vous aller pouvoir en choisir une.

 ,258 (32)
Vous avez appuyé sur la touche \11 enʃ
m'ouvrant.\0D++
Vous allez pouvoir changer l'applicationʃ
que je cherche et ouvre d'habitude.

 ,259 (32)
Désolé, je ne trouve plus l'application

 ,260 (32)
Désormais, je chercherais et ouvrirainʃ
automatiquement l'application

 ,261 (32)
Vous avez annulé. La prochaine fois, jeʃ
vous demanderais ++
quelle application vous désirez ouvrirʃ
automatiquement.

 ,262 (32)
.\0DPour changer d'application, appuyezʃ
sur la touche \11 ++
en m' uvrant.

 ,263 (32)
Désolé, je ne peux pas lire ce dont j'aiʃ
besoin; ++
je suis hors d'usage! Erreur N°

 ,264 (32)
Désolé, je ne peux pas enregistrer le nomʃ
de la nouvelle application.\0D

 ,265 (32)
Erreur N°

 ,266 (32)
Le disque est protégé en écriture
```

Source 'Appl Search.Asm'

```
* le symbole 'ʃ' indique la continuité de la
ligne

; programme de recherche et de lancement
; automatique d'application
;
; G. Zwingelstein
;

Include   Traps.D
Include   HFSMacs.txt
include   SysEqu.D
include   SysErr.D
include   FSEqu.D
include   PackMacs.Txt

Start  JSR     InitManagers

; Lit nom, type et créateur de l'application par
; défaut
; Function ReadApplHdles: OSErr;
        CLR     -(SP)
        JSR     ReadApplHdles
        MOVE    (SP)+,D0
        BEQ     @1

; Erreur lecture ressources
; Procédure ShowAlert(StrID1, StrID2, StrID3:
;    Integer; ErrNum: Integer);
        MOVE    #263,-(SP)
        CLR     -(SP)
        CLR     -(SP)
        MOVE    D0,-(SP)
        JSR     ShowAlert
        BRA     Fin

; Teste l'appui sur la touche option
; Function TestKey: Boolean;
@1      CLR.B   -(SP)
        JSR     TestKey
        TST.B   (SP)+
        BEQ     noOption

; Change le nom et le créateur de l'application par
; défaut
; Procédure ShowAlert(StrID1, StrID2, StrID3:
;    Integer; ErrNum: Integer);
        MOVE    #258,-(SP)
        CLR     -(SP)
        CLR     -(SP)
        CLR     -(SP)
        JSR     ShowAlert
; Function GetApplHdles: Boolean;
        CLR.B   -(SP)
        JSR     GetApplHdles
        TST.B   (SP)+
        BNE     ChangeName
; Annulation: efface le nom
; Procédure SetString(h: Handle; theString:
;    Str255);
```

```
        MOVE.L  ApplNameHdle(A5),-(SP)
        PEA     NullString
        _SetString
        BRA     ChangeName

; le nom existe-t-il ?
noOption
        MOVE.L  ApplNameHdle(A5),A0
        MOVE.L  (A0),A0
        MOVE.B  (A0),D0
        BNE     aName

; Pas de nom par défaut : demande un nom
; Procédure ShowAlert(StrID1, StrID2, StrID3:
;   Integer; ErrNum: Integer);
        MOVE    #257,-(SP)
        CLR     -(SP)
        CLR     -(SP)
        CLR     -(SP)
        JSR     ShowAlert
; Function GetApplHdles: Boolean;
        CLR.B   -(SP)
        JSR     GetApplHdles
        TST.B   (SP)+
        BEQ     Fin ; Annule

; Enregistre le nouveau nom
; Function WriteApplHdles: OSErr;
ChangeName
        CLR     -(SP)
        JSR     WriteApplHdles
        MOVE    (SP)+,D0
        BEQ     RetourFinder

; Erreur pendant l'enregistrement
; Procédure ShowAlert(StrID1, StrID2, StrID3:
;   Integer; ErrNum: Integer);
        MOVE    #264,-(SP)
        CMP     #wPrErr,D0
        BNE     @1
; disquette protégée en écriture
        CLR     D0
        MOVE    #266,-(SP)
        BRA     @2
; autres erreurs
@1      CLR     -(SP)
@2      CLR     -(SP)
        MOVE    D0,-(SP)
        JSR     ShowAlert
        BRA     Fin

RetourFinder
; retourne au bureau avec message sur
; l'application choisie
        MOVE.L  ApplNameHdle(A5),A0
        MOVE.L  (A0),A0
        MOVE.B  (A0),D0
        BNE     @1
; pas d'application
; Procédure ShowAlert(StrID1, StrID2, StrID3:
; Integer; ErrNum: Integer);
        MOVE    #261,-(SP)
        CLR     -(SP)
        CLR     -(SP)
        CLR     -(SP)
        JSR     ShowAlert
        BRA     Fin
; une application
; Procédure ShowAlert(StrID1, StrID2, StrID3:
;   Integer; ErrNum: Integer);
@1      MOVE    #260,-(SP)
        MOVE    #256,-(SP)
        MOVE    #262,-(SP)
        CLR     -(SP)
        JSR     ShowAlert
Fin     JSR     SetWatchCursor
        RTS     ; On Quitte

; Il faut chercher et lancer l'application
aName
; Procédure HLock(h: Handle);
        MOVE.L  ApplNameHdle(A5),A0
        _HLock
; Procédure HLock(h: Handle);
        MOVE.L  ApplTypesHdle(A5),A0
        _HLock

; Cherche l'application
; Fonction GetApplVRefNum (ApplNamePtr:
;   StringPtr; fTypes: Array[0..1] of OSType;
;   var Annule: Boolean): Integer;
        CLR     -(SP)
        MOVE.L  ApplNameHdle(A5),A0
        MOVE.L  (A0),-(SP)
        MOVE.L  ApplTypesHdle(A5),A0
        MOVE.L  (A0),-(SP)
        PEA     Annule(A5)
        JSR     GetApplVRefNum
        MOVE    (SP)+,ApplVRefNum(A5)
        BNE     @1
        TST.B   Annule(A5)
        BNE     Fin

; Impossible de trouver l'application sur un des
; volumes
; Procédure ShowAlert(StrID1, StrID2, StrID3:
;   Integer; ErrNum: Integer);
        MOVE    #259,-(SP)
        MOVE    #256,-(SP)
        MOVE    #262,-(SP)
        CLR     -(SP)
        JSR     ShowAlert
        BRA     Fin
@1
; Lance l'application trouvée
        JSR     SetWatchCursor
; Function SetVol(volName: StringPtr; vRefNum:
;   Integer): OSErr;
        CLR     -(SP)
        CLR.L   -(SP)
        MOVE    ApplVRefNum(A5),-(SP)
        JSR     SetVol
        MOVE    (SP)+,D0

; Procédure Launch
        MOVE    #0,-(SP)
        MOVE.L  ApplNameHdle(A5),A0
        MOVE.L  (A0),-(SP)
        MOVE.L  SP,A0
        _Launch

; chaîne de longueur nulle
NullString      DC      0

;-----------------------------------
;   InitManagers
;-----------------------------------

; Procédure InitAll;
; Initialise les managers
InitManagers
        PEA     -4(A5)
        _InitGraf
        _InitFonts
        MOVE.L  #$0000FFFF,D0
        _FlushEvents
        _InitWindows
        _InitMenus
        CLR.L   -(SP)
        _InitDialogs
        _TEInit
        _InitCursor
        RTS

;-----------------------------------
;   SetWatchCursor
;-----------------------------------

; Procédure SetWatchCursor;
; Curseur en forme de montre
SetWatchCursor
; Function GetCursor(cursorID: Integer):
;   CursHandle;
        CLR.L   -(SP)
        MOVE    #watchCursor,-(SP)
        _GetCursor
        MOVE.L  (SP)+,A0

; Procédure SetCursor(crsr: Cursor);
        MOVE.L  (A0),-(SP)
        _SetCursor
        RTS

;-----------------------------------
;   TestKey
;-----------------------------------

; Function TestKey: Boolean;
; Teste appui sur la touche option
TestKey
        LINK    A6,#-16

; Procédure GetKeys(var theKeys: KeyMap);
        PEA     -16(A6)
        _GetKeys

; Function BitTst(bytePtr: Ptr; bitNum:
;   LongInt): Boolean;
        CLR.B   -(SP)
        PEA     -16(A6)
        MOVE.L  #48,-(SP)
        _BitTst
        MOVE.B  (SP)+,8(A6)
        UNLK    A6
        RTS

;-----------------------------------
;   ShowAlert
;-----------------------------------
```

Un 'bug' du Microsoft Basic Compiler

Si votre compilateur vous affiche, sans raison apparente, le message :

`System error 43: File not found`

inutile de chercher une erreur de syntaxe dans le fichier source : le nom dudit fichier contient très probablement un caractère accentué...

```
; Procédure ShowAlert(StrID1, StrID2, StrID3:
;    Integer; ErrNum: Integer);
; Affiche la StopAlert d'ID 128 avec message d'ID
; donné et de No donné (si non nul)
ShowAlert
        LINK        A6,#-268
        MOVEM.L     D3/A3,-(SP)
        _InitCursor
; Numéro de l'erreur
        MOVE        8(A6),D0
        BEQ         @1
; Procédure NumToString(theNum: LongInt; var
;    theString: Str255);
        EXT.L       D0
        LEA         -256(A6),A0
        _NumToString
        BRA         @2
@1      CLR.B       -256(A6)
; Prépare les chaines de l'alerte pour ParamText
@2      MOVEQ       #4,D3
        LEA         -268(A6),A3
; Boucle des chaînes de l'alerte
@3      MOVE        10(A6,D3),D0
        BEQ         @4
; Récupère la chaine d'ID donné
; Function GetString(stringID: Integer):
StringHandle;
        CLR.L       -(SP)
        MOVE        D0,-(SP)
        _GetString
        MOVE.L      (SP)+,A0
        MOVE.L      A0,(A3)+
; prépare pour paramText
        MOVE.L      (A0),-(SP)
; Procédure HLock(h: Handle);
        _HLock
        BRA         @5
; chaîne vide
@4      CLR.L       (A3)+
        CLR.L       -(SP)
; dernière chaîne ?
@5      SUBQ        #2,D3
        BPL         @3

; Insère les chaînes dans l'alerte
; Procédure ParamText(param0, param1,
;    param2, param3: Str255);
; les 3 premiers paramètres sont déjà sur la pile
        PEA         -256(A6)
        _ParamText
; UnLock les chaînes de l'alerte
        MOVEQ       #2,D3
; Boucle de Unlock des chaînes de l'alerte
@6      MOVE.L      -(A3),D0
        BEQ         @7
; Procédure HUnLock(h: Handle);
        MOVE.L      D0,A0
        _HUnLock
; dernière chaîne ?
@7      DBRA        D3,@6
; Affiche la fenêtre d'alerte expliquant l'anomalie
; Procédure StopAlert(alertID: Integer;
;    filterproc: ProcPtr): Integer;
        CLR         -(SP)
        MOVE        #128,-(SP)
        CLR.L       -(SP)
        _StopAlert
        MOVE        (SP)+,D0
; Retour
        MOVEM.L     (SP)+,D3/A3
        UNLK        A6
        MOVE.L      (SP)+,A0
        ADDA        #8,SP
        JMP         (A0)

;-----------------------------------------
;   ReadApplHdles
;-----------------------------------------

; Function ReadApplHdles: OSErr
; lit le nom et le créateur de l'application par
; défaut à lancer
ReadApplHdles
        LINK        A6,#0
; Function GetString(stringID: Integer):
;    StringHandle;
        CLR.L       -(SP)
        MOVE        #256,-(SP)
        _GetString
        MOVE.L      (SP)+,ApplNameHdle(A5)
        BEQ         @1
; Function GetResource(theType: ResType;
;    theID: Integer): Handle;
        CLR.L       -(SP)
        MOVE.L      #'GNRL',-(SP)
        MOVE        #256,-(SP)
        _GetResource
        MOVE.L      (SP)+,ApplTypesHdle(A5)
        BEQ         @1
@1

; Function ResError: OSErr;
        CLR         -(SP)
        _ResError
        MOVE        (SP)+,8(A6)
        UNLK        A6
        RTS

;-----------------------------------------
;   GetApplHdles
;-----------------------------------------

; Function GetApplHdles: Boolean;
; récupère un nom de fichier par le dialogue
; standard de SFGetFile
GetApplHdles
        LINK        A6,#-(90)

; Procédure SFGetFile(where: Point; prompt:
;    Str255;fileFilter: ProcPtr; numTypes:
;    Integer; typeList: SFTypeList; dlgHook:
;    ProcPtr; var reply: SFReply);
        MOVE        #80,-(SP)
        MOVE        (SP),-(SP)
        CLR.L       -(SP)
        PEA         ApplFilter
        MOVE        #1,-(SP)
        PEA         APPLType
        CLR.L       -(SP)
        PEA         -74(A6)
        _SFGetFile
        MOVE.B      -74+rGood(A6),-8(A6)
        BEQ         @1
; Sauve le nom
; Procédure SetString(h: Handle; theString:
;    Str255);
        MOVE.L      ApplNameHdle(A5),-(SP)
        PEA         -74+rName(A6)
        _SetString
; sauve le type et créateur
; Function GetFInfo(fileName: Str255; vRefNum:
;    Integer; var fndrInfo:FInfo): OSErr;
        CLR         -(SP)
        PEA         -74+rName(A6)
        MOVE        -74+rVolume(A6),-(SP)
        PEA         -(74+16)(A6)
        JSR         GetFInfo
        MOVE        (SP)+,D0
        LEA         -(74+16)(A6),A0
        MOVE.L      ApplTypesHdle(A5),A1
        MOVE.L      (A1),A1
        MOVE.L      (A0)+,(A1)+
        MOVE.L      (A0)+,(A1)+
@1      UNLK        A6
        RTS

ApplType    DC      'APPL'

;-----------------------------------------
;   ApplFilter
;-----------------------------------------

; Function ApplFilter(paramBlock: ParmBlkPtr):
;    Boolean;
; Filtre les applications sans créateur pour
; SFGetFile
ApplFilter
        SF          8(SP)
        MOVE.L      4(SP),A0
        TST.L       ioFlUsrWds+fdCreator(A0)
        BNE         @1
        ST          8(SP)
@1      MOVE.L      (SP)+,(SP)
        RTS

;-----------------------------------------
;   WriteApplHdles
;-----------------------------------------

; Function WriteApplHdles: OSErr; marque les
;    ressources comme changées (elle seront
;    enregistrées en quittant)
WriteApplHdles
        LINK        A6,#0
; Procédure ChangedResource
;    (theResource:Handle);
        MOVE.L      ApplNameHdle(A5),-(SP)
        _ChangedResource
; Function ResError: Integer;
        CLR         -(SP)
        _ResError
        MOVE        (SP)+,8(A6)
        BNE         @1
; Procédure ChangedResource
;    (theResource:Handle);
        MOVE.L      ApplTypesHdle(A5),-(SP)
        _ChangedResource

; Function ResError: Integer;
        CLR         -(SP)
        _ResError
        MOVE        (SP)+,8(A6)
        BNE         @1
; Procédure SetResPurge(install: Boolean);
        ST          -(SP)
        _SetResPurge
@1      UNLK        A6
        RTS

;-----------------------------------------
;   GetApplVRefNum
;-----------------------------------------

; Fonction GetApplVRefNum (ApplNamePtr:
;    StringPtr; fTypes: Array[0..1] of OSType;
;    var Annule: Boolean): Integer;
; Cherche le fichier donné avec le type et créateur
; donné retourne un volume ou working directory
; reference number ou 0 si non trouvé

GetApplVRefNum
        LINK        A6,#-(ioHVQElSize+8)
```

```
        MOVEM.L  D3-D4,-(SP)
        CLR      20(A6)

; Initialise dialog d'annulation
        MOVE.L   8(A6),A0
        SF       (A0)
; Function GetString(stringID: Integer)
;   StringHandle;
        CLR.L    -(SP)
        MOVE     #262,-(SP)
        _GetString
        MOVE.L   (SP)+,A0
        MOVE.L   A0,-(ioHVQElSize+8)(A6)
; Procédure HLock(h: Handle);
        _HLock
; Procédure ParamText(param0, param1,
;   param2, param3: Str255);
        MOVE.L   16(A6),-(SP)
        MOVE.L   -(ioHVQElSize+8)(A6),A0
        MOVE.L   (A0),-(SP)
        CLR.L    -(SP)
        CLR.L    -(SP)
        _ParamText
; Procédure HUnLock(h: Handle);
        MOVE.L   -(ioHVQElSize+8)(A6),A0
        _HUnLock
; Function GetNewDialog(dialogID: Integer;
;   dStorage: Ptr; Behind: WindowPtr);
;   DialogPtr;
        CLR.L    -(SP)
        MOVE     #129,-(SP)
        CLR.L    -(SP)
        MOVE.L   #-1,-(SP)
        _GetNewDialog
        MOVE.L   (SP)+,-(ioHVQElSize+4) ∫
                 (A6)
; met curseur flèche
        _InitCursor
; Boucle pour chaque volume
        CLR      D3
; Volume suivant
NextVolume
; Gestion annulation
; Function GereAnnule: Boolean;
        CLR.B    -(SP)
        MOVE.L   A6,-(SP)
        JSR      GereAnnule
        TST.B    (SP)+
        BNE      NotFounded
;Pas d'annulation : volume suivant
        ADD      #1,D3
; Système HFS ? (cf inside Macintosh IV-97)
        TST      FSFCBLen
        BPL      SystemHFS

; Système non HFS

; Function PBGetVInfo(paramBlock: ParmBlkPtr;
;   async: Boolean): OSErr;
        LEA      -ioHVQElSize(A6),A0
        CLR.L    ioVNPtr(A0)
        CLR      ioVRefNum(A0)
        MOVE     D3,ioVolIndex(A0)
        _GetVolInfo
        BNE      NotFounded
        MOVE     ioVRefNum(A0),D4
        BRA      MFSVol

; Système HFS

SystemHFS
; Function PBHGetVInfo (paramBlock:
;   HParmBlkPtr; async: Boolean): OSErr;
        LEA      -ioHVQElSize(A6),A0
        CLR.L    ioVNPtr(A0)
        CLR      ioVRefNum(A0)
        MOVE     D3,ioVolIndex(A0)
        _HGetVInfo
        BNE      NotFounded
; volume HFS ou MFS (ou autre) ?
        MOVE     ioVRefNum(A0),D4
        CMPI     #SigWord,ioVSigWord(A0)
        BEQ      MFSVol
        CMPI     #TSigWord,ioVSigWord∫
                 (A0)
        BEQ      HFSVol
        BRA      NextVolume

; Cherche l'application sur le volume MFS volume
MFSVol
; Function FindAppl(theVRefNum: Integer);
OSErr;
        CLR      -(SP)
        MOVE     D4,-(SP)
        MOVE.L   A6,-(SP)
        JSR      FindAppl
        TST      (SP)+
        BNE      NextVolume
        BRA      Founded

; HFS volume
HFSVol
; Fonction FindApplWDRefNum (ioVRefNum:
;   Integer; ioDirID: LongInt): Integer;
        CLR      -(SP)
        MOVE     D4,-(SP)
        MOVE.L   #2,-(SP)
        MOVE.L   A6,-(SP)
        JSR      FindApplWDRefNum
        MOVE     (SP)+,D4
        BEQ      NextVolume

Founded
        MOVE     D4,20(A6)

;Fin
NotFounded

; Procédure DisposDialog (theDialog: DialogPtr);
        MOVE.L   -(ioHVQElSize+4)(A6),∫
                 -(SP)
        _DisposDialog
        MOVEM.L  (SP)+,D3-D4
        UNLK     A6
        MOVE.L   (SP)+,A0
        ADDA     #12,SP
        JMP      (A0)

;-----------------------------------------
;   FindAppl
;-----------------------------------------
; Function FindAppl(theVRefNum: Integer);
;   OSErr;
; Cherche existence de l'application donnée sur
; volume MFS
; !!! Attention : Fonction interne à la Fonction
; GetApplVRefNum donc pointeur sur variable
; locale en plus sur la pile
FindAppl
        LINK     A6,#-ioFQElSize
; Function PBGetFInfo(paramBlock: ParmBlkPtr;
;   async: Boolean); OSErr;
        LEA      -ioFQElSize(A6),A0
        MOVE.L   8(A6),A1
        MOVE.L   16(A1),ioFileName(A0)
        MOVE     12(A6),ioVRefNum(A0)
        CLR      ioFileType(A0)
        CLR      ioFDirIndex(A0)
        _GetFileInfo
        MOVE     D0,14(A6)
        BNE      @2
; Check type et créateur
; Function CkeckInfo (fndrInfo:FInfo): Boolean
        CLR.B    -(SP)
        PEA      ioFIUsrWds(A0)
        MOVE.L   8(A6),-(SP)
        JSR      CheckInfo
        TST.B    (SP)+
        BNE      @2
; Pas le bon type (File not found)
@1      MOVE     #InfErr,14(A6)
@2      UNLK     A6
        MOVE.L   (SP)+,A0
        ADDA     #6,SP
        JMP      (A0)

;-----------------------------------------
;   FindApplWDRefNum
;-----------------------------------------

; Fonction FindApplWDRefNum (ioVRefNum:
;   Integer; ioDirID: LongInt): Integer;
; Cherche existence de l'application donnée sur
; volume HFS et retourne le working directory
; reference number le contenant ou 0 sinon
; !!! Attention : Fonction interne à la Fonction
; GetApplVRefNum donc pointeur sur variable
; locale en plus sur la pile
FindApplWDRefNum
```

Copies d'écran Minitel, n° 27

Paru il y a un an, notre programme Minitel du n° 27 semble avoir encore beaucoup de succès. Vous êtes cependant nombreux à 'buter' sur le mode Copie d'écran. Voici résumée la procédure d'enregistrement :

1. mettre l'ordinateur en mode réception de copies d'écran ;
2. depuis le Minitel, consulter les pages Vidéotex ;
3. lorsqu'un écran doit être enregistré sur l'ordinateur et **seulement** lorsqu'il est affiché sur le Minitel, presser RETURN (sur l'Apple *II*) ou cliquer Réception de copies (sur le Macintosh) ;
4. faire FNCT-I suivi de F (le Minitel **doit** être doté de la touche FNCT).

Reprendre à la phase 3 pour chaque nouvelle page : les phases 3 et 4 doivent s'enchaîner sans changer de page Minitel. Si vous avez le serveur envoie le message *Caractère ";"erroné*, faites FNCT-P suivi de I, au début de l'ensemble des copies d'écran.

```
        LINK    A6,#-ioHFQElSiz
        CLR     18(A6)
; Function PBHGetFInfo (paramBlock:
;   HParmBlkPtr; async: Boolean); OSErr;
        LEA     -ioHFQElSiz(A6),A0
        MOVE.L  8(A6),A1
        MOVE.L  16(A1),ioFileName(A0)
        MOVE    16(A6),ioVRefNum(A0)
        CLR     ioFDirIndex(A0)
        MOVE.L  12(A6),ioDirID(A0)
        _HGetFileInfo
        BNE     @1
; Check type et créateur
; Function CkeckInfo (fndrInfo:FInfo): Boolean
        CLR.B   -(SP)
        PEA     ioFIUsrWds(A0)
        MOVE.L  8(A6),-(SP)
        JSR     CheckInfo
        TST.B   (SP)+
        BEQ     @1
; Bon type et créateur
; Function GetWDRefNum (VRefNum: Integer;
;   WDDirID: LongInt): Integer;
        CLR     -(SP)
        MOVE    16(A6),-(SP)
        MOVE.L  12(A6),-(SP)
        JSR     GetWDRefNum
        MOVE    (SP)+,18(A6)
        BNE     @3
@1      LEA     -ioHFQElSiz(A6),A0
        CLR.L   ioFileName(A0)
        CLR     ioFDirIndex(A0)
        MOVE    16(A6),ioVRefNum(A0)

@2
; Gestion annulation
; Function GereAnnule: Boolean;
        CLR.B   -(SP)
        MOVE.L  8(A6),-(SP)
        JSR     GereAnnule
        TST.B   (SP)+
        BNE     @3

; Directory suivant
        LEA     -ioHFQElSiz(A6),A0
        ADD     #1,ioFDirIndex(A0)
        MOVE.L  12(A6),ioDrDirID(A0)
        _GetCatInfo
        BNE     @3
        BTST    #4,ioFIAttrib(A0)
        BEQ     @2
; Appelle la fonction pour le directory trouvé
; Fonction FindAppIWDRefNum (ioVRefNum:
;   Integer; ioDirID: LongInt): Integer;
        CLR     -(SP)
        MOVE    16(A6),-(SP)
        MOVE.L  ioDrDirID(A0),-(SP)
        MOVE.L  8(A6),-(SP)
        BSR     FindAppIWDRefNum
        MOVE    (SP)+,18(A6)
        BEQ     @2
@3      UNLK    A6
        MOVE.L  (SP)+,A0
        ADDA    #10,SP
        JMP     (A0)

;-----------------------------------
;   CheckInfo
;-----------------------------------

; Function CkeckInfo (fndrInfo:FInfo): Boolean
; Vérifie le type et créateur du fichier
; !!! Attention : Fonction interne à la Fonction
; GetAppIVRefNum donc pointeur sur variable
; locale en plus sur la pile
CheckInfo
        SF      12(SP)
;Check type
        MOVE.L  8(SP),A0
        MOVE.L  4(SP),A1
        MOVE.L  12(A1),A1
        CMP.L   (A0)+,(A1)+
        BNE     @1
        CMP.L   (A0)+,(A1)+
        BNE     @1
; bon type et créateur
        ST      12(SP)
@1      MOVE.L  (SP)+,A0
        ADDA    #8,SP
        JMP     (A0)

;-----------------------------------
; GereAnnule
;-----------------------------------

; Function GereAnnule: Boolean;
; Gère les événements pendant la recherche de
; l'application
; !!! Attention : Fonction interne à la Fonction
; GetAppIVRefNum donc pointeur sur variable
; locale en plus sur la pile
GereAnnule
        LINK    A6,#-(evtBlkSize+6)

; Annulation déjà effectuée ?
        MOVE.L  8(A6),A0
        MOVE.L  8(A0),A0
        MOVE.B  (A0),12(A6)
        BNE     @2

; Function GetNextEvent (eventMask: Integer; var
;   theEvent: EventRecord): Boolean;
@1      CLR.B   -(SP)
        MOVE    #-1,-(SP)
        PEA     -evtBlkSize(A6)
        _GetNextEvent
        TST.B   (SP)+
        BEQ     @2
; Function IsDialogEvent(theEvent: EventRecord):
;   Boolean;
        CLR.B   -(SP)
        PEA     -evtBlkSize(A6)
        _IsDialogEvent
        TST.B   (SP)+
        BEQ     @2
; Function DialogSelect(theEvent: EventRecord;
;   var theDialog: DialogPtr; var itemHit
;   Integer): Boolean;
        CLR.B   -(SP)
        PEA     -evtBlkSize(A6)
        PEA     -(evtBlkSize+4)(A6)
        PEA     -(evtBlkSize+6)(A6)
        _DialogSelect
        MOVE.L  8(A6),A0
        MOVE.L  8(A0),A0
;Variable d'annulation
        MOVE.B  (SP),(A0)
        MOVE.B  (SP)+,12(A6)
        BEQ     @1
@2      UNLK    A6
        MOVE.L  (SP)+,A0
        ADDA    #4,SP
        JMP     (A0)

;-----------------------------------
;   GetWDRefNum
;-----------------------------------

; Function GetWDRefNum (VRefNum: Integer;
;   WDDirID: LongInt): Integer;
; retourne le working directory reference number
; du WDDirID ou 0 si erreur

GetWDRefNum
        LINK    A6,#-ioHFQElSiz
        CLR     14(A6)
        LEA     -ioHFQElSiz(A6),A0
        CLR.L   ioFileName(A0)
        MOVE    12(A6),ioVRefNum(A0)
        CLR.L   ioWDProcID(A0)
        MOVE.L  8(A6),ioWDDirID(A0)
        _OpenWD
        BNE     @1
        MOVE    ioVRefNum(A0),14(A6)
@1      UNLK    A6
        MOVE.L  (SP)+,A0
        ADDA    #6,SP
        JMP     (A0)

;-----------------------------------
;   GetFInfo
;-----------------------------------

; Function GetFInfo(fileName: Str255; vRefNum:
;   Integer; var fndrInfo:FInfo): OSErr;
; Récupère les infos du finder pour le fichier
; donné, sur le volume donné
GetFInfo
        LINK    A6,#-ioFQElSize
        LEA     -ioFQElSize(A6),A0
        MOVE.L  14(A6),ioFileName(A0)
        MOVE    12(A6),ioVRefNum(A0)
        CLR     ioFileType(A0)
        CLR     ioFDirIndex(A0)
        _GetFileInfo
        MOVE    D0,18(A6)

; Procédure BlockMove(sourcePtr, destPtr: Ptr;
;   byteCount: Size);
        LEA     ioFIUsrWds(A0),A0
        MOVE.L  8(A6),A1
        MOVEQ   #16,D0
        _BlockMove
        UNLK    A6
        MOVE.L  (SP)+,A0
        ADDA    #10,SP
        JMP     (A0)

;-----------------------------------
;   SetVol
;-----------------------------------

; Function SetVol(volName: StringPtr; vRefNum:
;   Integer): OSErr;
; met le volume ou working directory par défaut

SetVol
        LINK    A6,#-ioQElSize
        LEA     -ioQElSize(A6),A0
        MOVE.L  10(A6),ioVNPtr(A0)
        MOVE    8(A6),ioVRefNum(A0)
        _SetVol
        MOVE    D0,14(A6)
        UNLK    A6
        MOVE.L  (SP)+,A0
        ADDA    #6,SP
        JMP     (A0)

;-----------------------------------
;   Variable globales
;-----------------------------------

AppINameHdle    DS  4
AppITypesHdle   DS  4
AppIVRefNum     DS  2
Annule          DS  2
```

Booter dans un dossier ? NEWBOOT

Patrick Guyou

Le fichier "Basic. System" lance éternellement l'exécution du fichier *Startup* s'il le trouve dans le catalogue principal du volume, ce *Startup* pouvant être aussi bien Basic que binaire. Il est simple, sinon pratique, de modifier ce nom à l'aide d'un éditeur type Copy][+, Bag of Tricks, mais la manipulation reste inconfortable.

NewBoot est une modification de Basic.System qui permet de changer ce nom à volonté : il suffit de presser la barre d'espacement lors du boot. Un écran apparaît alors, donnant le nom du volume de boot, le nom du programme lancé par Basic. System et vous proposant d'entrer un autre chemin d'accès pour le démarrage.

À ce niveau, entrer simplement le nom d'un programme ou bien un chemin d'accès complet. Il est possible d'indiquer :

/DISK/DIR1/DIR2/DIR3/DIR4/PROGRAMME

ainsi le fichier 'Programme' situé au 5ème niveau du volume sera exécuté au démarrage (le préfixe sera par la même occasion établi au niveau du programme lancé).

Le programme lancé automatiquement par Basic.System peut également se trouver sur un autre volume : lorsque pour quelques temps, il est prévu d'utiliser fréquemment AppleWorks il devient aisé de préciser :

/U3.5/APPLEWORKS/APLWORKS.SYSTEM

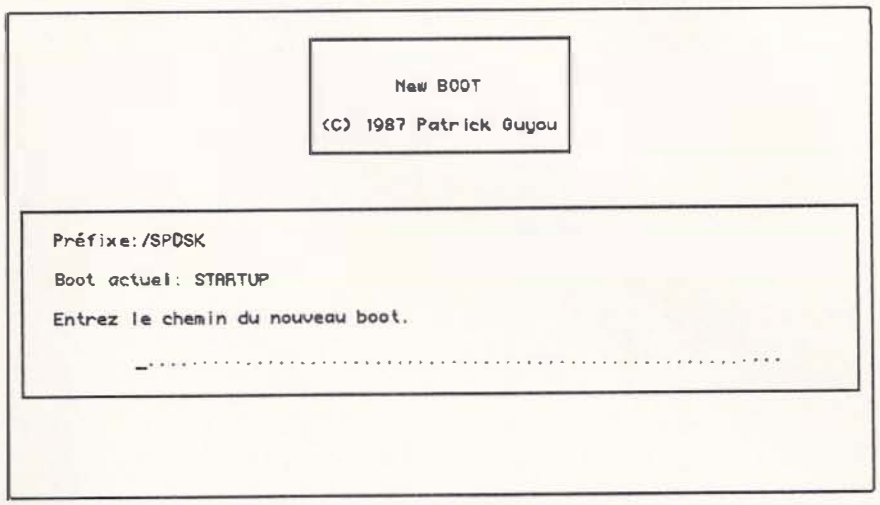
Utilisation

Saisir et sauvegarder *BOOT* et *PATCH*, mettre ces deux fichiers sur le volume comportant le Basic.System à modifier, faire :

- *PATCH*

et attendre un instant.

Maintenant pour changer de startup, la barre d'espace remplace Copy][+...



Récapitulation BOOT

Après avoir saisi cette récapitulation sous moniteur, vous la sauvegarderez par : BSAVE BOOT, A$4800, L1984

```
4800:AD 00 C0 C9 A0 F0 35 20
4808:58 FC AD 07 20 C9 2F F0
4810:03 4C 47 20 AE 06 20 BD
4818:07 20 C9 2F F0 04 CA 4C
4820:17 48 A0 00 CA BD 07 20
4828:9D 40 4B C8 E0 00 F0 03
4830:4C 24 48 8C 3F 4B 20 36
4838:49 4C 47 20 A9 07 8D F2
4840:03 A9 48 8D F3 03 49 A5
4848:8D F4 03 A9 00 A2 FF 9A
4850:2C 10 C0 20 00 C3 A0 00
4858:20 34 4A A0 00 20 E3 49
4860:A0 01 20 34 4A 20 64 49
4868:A0 01 20 E3 49 48 A9 0C
4870:20 5B FB A9 12 85 24 68
4878:AE 06 20 A0 00 B9 07 20
4880:49 80 20 ED FD C8 CA F0
4888:03 4C 7D 48 A0 02 20 E3
4890:49 48 A9 10 8D 89 4A A9
4898:0D 8D 8A 4A A9 35 8D 88
48A0:4A 20 8B 4A 68 AD 3F 4B
48A8:8D 06 20 A2 00 BD 40 4B
48B0:38 E9 80 9D 07 20 E8 EC
48B8:06 20 F0 03 4C AD 48 A9
48C0:00 9D 07 20 E8 9D 07 20
48C8:A9 E3 8D 62 4F 20 56 4F
48D0:20 F4 48 A9 21 8D 62 4F
48D8:20 56 4F 20 58 FC A0 0D
48E0:20 E3 49 20 36 4B C9 8D
48E8:D0 F9 AD 8B C0 AD 8B C0
48F0:4C 03 BF 00 20 00 BF C8
48F8:19 49 B0 1A AD 1E 49 8D
4900:20 49 8D 28 49 20 00 BF
4908:CB 1F 49 B0 09 20 00 BF
4910:CC 27 49 B0 01 60 4C A0
4918:4B 03 29 49 00 08 00 04
4920:00 00 20 58 2F 00 00 01
4928:00 0C 42 41 53 49 43 2E
4930:53 59 53 54 45 4D A5 43
4938:8D 56 49 20 00 BF C6 44
4940:49 B0 04 60 01 3F 4B AD
4948:56 49 8D 61 49 20 57 49
4950:A2 FF 9A 4C 47 20 00 20
4958:00 BF C5 60 49 B0 B7 60
4960:02 00 00 08 20 00 BF C7
4968:70 49 B0 DB 20 73 49 60
4970:01 92 49 48 A9 0A 20 5B
4978:FB A9 0E 85 24 68 A2 00
4980:BD 93 49 49 80 20 ED FD
4988:E8 EC 92 49 F0 03 4C 80
4990:49 60 00 00 00 00 00 00
4998:00 00 00 00 00 00 00 00
49A0:00 00 00 00 00 00 00 00
49A8:00 00 00 00 00 00 00 00
49B0:00 00 00 00 00 00 00 00
49B8:00 00 00 00 00 00 00 00
49C0:00 00 00 00 00 00 00 00
49C8:00 00 00 00 00 00 00 00
49D0:00 00 00 00 00 00 00 00
49D8:00 00 00 00 00 00 00 00
```

```
49E0:00 00 00 B9 18 4A 85 06
49E8:B9 26 4A 85 07 A0 00 B1
49F0:06 20 5B FB C8 B1 06 85
49F8:24 C8 F0 16 B1 06 C9 8D
4A00:F0 0A C9 00 F0 11 20 ED
4A08:FD 4C F9 49 C8 F0 03 4C
4A10:EF 49 E6 07 4C FC 49 60
4A18:A8 D7 E6 3C 5D 7E 9F C0
4A20:E1 02 23 44 5B 0A 4E 4E
4A28:4E 4C 4C 4C 4C 4C 4C 4D
4A30:4D 4D 4D 4F AD 1E C0 30
4A38:03 8D 0F C0 B9 84 4A 85
4A40:06 B9 86 4A 85 07 A9 3F
4A48:85 32 A0 00 B1 06 20 5B
4A50:FB EA C8 B1 06 85 24 C8
4A58:F0 1B A9 1B 20 ED FD B1
4A60:06 C9 8D F0 0A C9 00 F0
4A68:11 20 F0 FD 4C 57 4A C8
4A70:F0 03 4C 4C 4A E6 07 4C
4A78:5A 4A A9 18 20 F0 FD A9
4A80:FF 85 32 60 70 D0 4D 4D
4A88:00 00 00 A2 00 AC 8A 4A
4A90:84 24 AD 89 4A 20 5B FB
4A98:A9 AE 20 ED FD EC 88 4A
4AA0:F0 04 E8 4C 98 4A A2 00
4AA8:AC 8A 4A 84 24 AD 89 4A
4AB0:20 5B FB AD 0D 4B 20 ED
4AB8:FD 20 36 4B C9 8D F0 4F
4AC0:C9 FF F0 32 C9 9B F0 56
4AC8:C9 88 F0 2A C9 A1 90 1C
4AD0:C9 FE B0 18 EC 88 4A B0
4AD8:E0 84 24 20 ED FD 9D 40
4AE0:4B AD 0D 4B 20 ED FD E8
4AE8:C8 4C B9 4A 5A DA 20 23
4AF0:4B FA 7A 4C B9 4A CC 8A
4AF8:4A F0 0F 88 CA 84 24 AD
4B00:0D 4B 20 ED FD A9 AE 20
4B08:ED FD 4C B9 4A DF 00 E0
4B10:03 90 D9 8E 3F 4B 84 24
4B18:A9 AE 20 ED FD 60 4C 3C
4B20:48 FF FF AE 21 4B AC 22
4B28:4B 2C 30 C0 88 D0 FD CA
4B30:CE 22 4B D0 F1 60 AD 00
4B38:C0 10 FB 2C 10 C0 60 00
4B40:00 00 00 00 00 00 00 00
4B48:00 00 00 D0 00 00 00 00
4B50:00 00 00 00 00 00 00 00
4B58:00 00 00 00 00 00 00 00
4B60:00 00 00 00 00 00 00 00
4B68:00 00 00 00 00 00 00 00
4B70:00 00 00 00 00 00 00 00
4B78:00 00 00 00 00 00 00 00
4B80:00 00 00 00 00 00 00 00
4B88:00 00 00 00 00 00 00 00
4B90:00 00 00 00 00 00 00 00
4B98:00 00 00 00 00 00 00 00
4BA0:18 C9 27 D0 05 A0 03 4C
4BA8:0B 4C C9 2B D0 05 A0 04
4BB0:4C 0B 4C C9 40 D0 05 A0
4BB8:05 4C 0B 4C C9 44 D0 05
4BC0:A0 06 4C 0B 4C C9 45 D0
4BC8:05 A0 07 4C 0B 4C C9 46
4BD0:D0 05 A0 08 4C 0B 4C C9
4BD8:48 D0 05 A0 09 4C 0B 4C
4BE0:C9 4E D0 05 A0 0A 4C 0B
4BE8:4C A0 0C 48 20 E3 49 A0
4BF0:0B 20 E3 49 68 20 DA FD
4BF8:20 23 4B 20 36 4B C9 9B
4C00:F0 06 20 23 4B 4C FB 4B
4C08:4C 25 4C 48 20 E3 49 68
4C10:20 23 4B A0 0C 20 E3 49
```

Tours Micro-Club...

...90, avenue Maginot – 37100 Tours, édite un bulletin bien sympathique : "Le p'tit pommier international". Il s'agit, selon ses auteurs, du premier journal édité en DOS 3.3 et dont les pages sont numérotées en hexadécimal...

```
4C18:20 36 4B C9 9B F0 06 20
4C20:23 4B 4C 18 4C A9 00 8D
4C28:94 BF 20 00 BF CC 37 4C
4C30:B0 07 A9 00 4C 3C 48 01
4C38:00 4C A0 4B 14 19 A0 A0
4C40:A0 C5 F2 F2 E5 F5 F2 A0
4C48:E4 A7 C5 CE D4 D2 C5 C5
4C50:AF D3 CF D2 D4 C9 C5 AE
4C58:AE AE A1 A0 00 14 19 C4
4C60:E9 F3 F1 F5 E5 A0 F0 F2
4C68:EF F4 FB E7 FB A0 E5 EE
4C70:A0 FB E3 F2 E9 F4 F5 F2
4C78:E5 AE AE AE A1 00 14 19
4C80:A0 A0 AD A0 A0 C5 F2 F2
4C88:E5 F5 F2 A0 E4 E5 A0 F3
4C90:F9 EE F4 E1 F8 E5 AE AE
4C98:AE A1 A0 A0 A0 A0 00 14
4CA0:19 A0 A0 A0 A0 A0 C3 E1
4CA8:F4 E1 EC EF E7 F5 E5 A0
4CB0:E9 EE E3 EF EE EE F5 AE
4CB8:AE AE A1 A0 A0 A0 A0 00
4CC0:14 19 A0 A0 A0 CE EF ED
4CC8:A0 E4 E5 A0 F6 EF EC F5
4CD0:ED E5 A0 E9 EE E3 EF EE
4CD8:EE F5 AE AE AE A1 A0 A0
4CE0:00 14 19 A0 A0 A0 A0 A0
4CE8:A0 C6 E9 E3 E8 E9 E5 F2
4CF0:A0 E9 EE E3 EF EE EE F5
4CF8:AE AE AE A1 A0 A0 A0 A0
4D00:A0 00 14 19 A0 A0 CC E1
4D08:A0 E4 E9 F3 F1 F5 E5 F4
4D10:F4 E5 A0 E5 F3 F4 A0 F0
4D18:EC E5 E9 EE E5 AE AE AE
4D20:A1 A0 00 14 19 A0 A0 A0
4D28:A0 C6 E9 E3 E8 E9 E5 F2
4D30:A0 F6 E5 F2 F2 EF F5 E9
4D38:EC EC FB AE AE AE A1 A0
4D40:A0 A0 A0 00 14 19 A0 A0
4D48:A0 A0 A0 A0 A0 A0 A0 C5
4D50:F2 F2 E5 F5 F2 AD CE DB
4D58:A0 A4 00 16 20 C1 F0 F0
4D60:F5 F9 E5 FA A0 F3 F5 F2
4D68:A0 BC C5 D3 C3 BE AE 00
4D70:01 1B 5A 4C 4C 4C 4C 4C
4D78:4C 4C 4C 4C 4C 4C 4C 4C
4D80:4C 4C 4C 4C 4C 4C 4C 4C
4D88:4C 4C 4C 5F 8D 02 1B 5A
4D90:8D 02 34 5F 8D 03 1B 5A
4D98:8D 03 34 5F 8D 04 1B 5A
4DA0:8D 04 34 5F 8D 05 1B 5A
4DA8:8D 05 34 5F 8D 06 1B 5A
4DB0:8D 06 34 5F 8D 07 1C 4C
4DB8:4C 4C 4C 4C 4C 4C 4C 4C
4DC0:4C 4C 4C 4C 4C 4C 4C 4C
4DC8:4C 4C 4C 4C 4C 4C 4C 00
4DD0:09 03 5A 4C 4C 4C 4C 4C
4DD8:4C 4C 4C 4C 4C 4C 4C 4C
4DE0:4C 4C 4C 4C 4C 4C 4C 4C
4DE8:4C 4C 4C 4C 4C 4C 4C 4C
4DF0:4C 4C 4C 4C 4C 4C 4C 4C
4DF8:4C 4C 4C 4C 4C 4C 4C 4C
4E00:4C 4C 4C 4C 4C 4C 4C 4C
4E08:4C 4C 4C 4C 4C 4C 4C 4C
4E10:4C 4C 4C 4C 4C 4C 4C 4C
4E18:4C 4C 4C 5F 8D 0A 03 5A
4E20:8D 0A 4C 5F 8D 0B 03 5A
4E28:8D 0B 4C 5F 8D 0C 03 5A
4E30:8D 0C 4C 5F 8D 0D 03 5A
4E38:8D 0D 4C 5F 8D 0E 03 5A
4E40:8D 0E 4C 5F 8D 0F 03 5A
4E48:8D 0F 4C 5F 8D 10 03 5A
4E50:8D 10 4C 5F 8D 11 03 5A
4E58:8D 11 4C 5F 8D 12 04 4C
4E60:4C 4C 4C 4C 4C 4C 4C 4C
4E68:4C 4C 4C 4C 4C 4C 4C 4C
4E70:4C 4C 4C 4C 4C 4C 4C 4C
4E78:4C 4C 4C 4C 4C 4C 4C 4C
4E80:4C 4C 4C 4C 4C 4C 4C 4C
4E88:4C 4C 4C 4C 4C 4C 4C 4C
4E90:4C 4C 4C 4C 4C 4C 4C 4C
4E98:4C 4C 4C 4C 4C 4C 4C 4C
4EA0:4C 4C 4C 4C 4C 4C 4C 00
4EA8:03 24 CE E5 F7 A0 C2 CF
4EB0:CF D4 8D 05 1D A8 C3 A9
4EB8:A0 B1 B9 B8 B7 A0 D0 E1
4EC0:F4 F2 E9 E3 EB A0 C7 F5
4EC8:F9 EF F5 8D 0A 06 D0 F2
4ED0:FB E6 E9 F8 E5 BA 00 0C
4ED8:06 C2 EF EF F4 A0 E1 E3
4EE0:F4 F5 E5 EC BA 00 0E 06
4EE8:C5 EE F4 F2 E5 FA A0 EC
4EF0:E5 A0 E3 E8 E5 ED E9 EE
4EF8:A0 E4 F5 A0 EE EF F5 F6
4F00:E5 E1 F5 A0 E2 EF EF F4
4F08:AE 00 0A 16 BC D2 E5 F4
4F10:F5 F2 EE BE A0 D0 EF F5
4F18:F2 A0 F1 F5 E9 F4 F4 E5
4F20:F2 A0 EC E5 A0 F0 F2 EF
4F28:E7 F2 E1 ED ED E5 AE 8D
4F30:0C 16 BC C3 F4 F2 EC AD
4F38:D2 E5 F3 E5 F4 BE A0 D0
4F40:EF F5 F2 A0 F2 E5 EC E1
4F48:EE E3 E5 F2 A0 EC E5 A0
4F50:E2 EF EF F4 AE 00 20 00
4F58:BF C3 5F 4F B0 0F 60 07
4F60:29 49 E3 FF 00 20 00 00
4F68:00 00 00 00 00 4C A0 4B
```

Sur la disquette Apple //

La revue Pom's est accompagnée opti nnellement d'une disquette 5,25' 140Ko ou 3,5' 800Ko. En 140Ko, le recto e t en format DOS, le verso en ProDOS. La disquette 800Ko est formatée ProDOS : 'Veiller à être sous le bon système d'exploitation avant d'exécuter un programme.
Pour pouvoir démarrer directement avec la disquette formatée en ProDOS, y copier ProDOS (les possesseurs de GS l'ont sous le n m P8) et Basic.System. Notre téléphone en cas de difficultés : (1) 39 51 24 43.

HGR.MNTL	ProDOS ou DOS 3.3 (RUN)
DENSITE	Utilitaire pour HGR.MNTL
DENSITE.S	Source Merlin-Pro
COMPACTEUR	ProDOS ou DOS 3.3 (RUN)
STARTUP.D	WPL pour AppleWriter ProDOS
VARIDATE.D	WPL
INIDATE.D	WPL
INIDATE.D	WPL
DATE	TXT contenant la date
STARTUP.N	WPL pour AppleWriter ProDOS
VARIDATE.N	WPL
INIDATE.N	WPL
CHOIX	WPL
NUMERO	WPL
GLOSSAIRE	WPL
COMMANDE	WPL
RELANCE	WPL
FICHRELANCE	TXT exemple
FICHCOMMANDE	TXT exemple
HEURE	WPL pour AppleWriter ProDOS
PATCH	ProDOS, programme à RUNer
BOOT.SOURCE	Source ProCODE
BOOT	patch
QD.BIN	Objet à exécuter depuis ProDOS 16
QD.SAMPLE	programme exemple (RUN ou –)
QD.MAIN	Source Apw
QD.SRCE	Source Apw
ONLINE.EXE	Objet à BRUNer sous ProDOS
ONLINE.S	Source Merlin-Pro
CAT & MOUSE	Objet à BRUNer sous DOS 3.3
SOURCE.LISA	Source Lisa
COMMANDE.CLV	Bon de commande CalvaCom

Programme 'Patch'

ProDOS

Ce patch fonctionne sur Basic.System version 1.1. Pour connaître le numéro de votre version, renommez votre programme de startup et bootez.

```
10  REM  ************************
20  REM  *                      *
30  REM  *  Programme de Patch  *
40  REM  *                      *
50  REM  *     BASIC.SYSTEM     *
60  REM  *                      *
70  REM  ************************
80  REM
90 D$ =  CHR$ (4)
100  PRINT D$"BLOAD BASIC.SYSTEM,T
   SYS,A$2000"
110  PRINT D$"BLOAD BOOT"
120  POKE 8193,0: POKE 8194,72
130  PRINT D$"UNLOCK BASIC.SYSTEM"
140  PRINT D$"BSAVE BASIC.SYSTEM,T
   SYS,A$2000,L12150"
150  PRINT D$"LOCK BASIC.SYSTEM"
160  HOME
170  PRINT "OK, C'EST FAIT..."
```

Source BOOT.SOURCE
Assembleur ProCODE

][+
//e
//e+
//c
][gs

```
*****************************************
*                                       *
*                New Boot               *
*                                       *
*          (C) 1987 Patrick Guyou       *
*                                       *
*               Version 1.0             *
*                                       *
*    Apple //e+ - Apple //c - Apple //gs *
*                                       *
*        65c02 - Assembleur procode     *
*                                       *
*****************************************
*
*   NEW.BOOT
*   --------
*   Programme de patch à Basic.system permettant de changer
*   le nom du programme 'booté' au démarrage du disque.
*   Le nouveau boot peut s'effectuer, non seulement à différents
*   niveaux du catalogue, mais aussi sur un autre disque.
*   Une disquette 5" 1/4 Pourra booter le programme se trouvant au
*   5ème niveau d'un disque dur.
*   " /DISQUE/DIR1/DIR2/DIR3/DIR4/DIR5/PROGRAMME ", est correct
*
*   Attention:
*   ---------
*   La syntaxe : "/DISQUE/DIR/"  seule, entrainerait un "FILE TYPE M
                                 ISMATCH"
*   lors du boot, mais le prefixe serait établi sur "/DISQUE/DIR/".
*
*
TRANS    EQU  $06        ;Pour adressage indirect page 0
TABV     EQU  $FB5B      ;Tabulation verticale
TABH     EQU  $24        ; "    "    horizontale
MEM.SD   EQU  $43        ; slot/drive actif
COL80    EQU  $C300      ;passage en 80 Col.
ALTOFF   EQU  $C00E      ; commutateurs caractéres
ALTSET   EQU  $C00F      ;     "          "
ALTFL    EQU  $C01E
HOME     EQU  $FC58      ;efface écran
COUT     EQU  $FDED      ;affiche caractére accumulateur
COUT1    EQU  $FDF0
KBD      EQU  $C000      ;touche appuyée
STROBE   EQU  $C010      ;mise à zéro clavier
MLI      EQU  $BF00      ;entrée MLI
HP       EQU  $C030      ;stimulation haut-parleur
```

```
BASIC.R  EQU  $03D0      ;retour basic
PRBYTE   EQU  $FDDA      ;affiche code hexa accumulateur
LEVEL    EQU  $BF94      ;
LCHAINE  EQU  $2006      ;adresse 'basic.system'
CHAINE   EQU  $2007      ;   "         "
BOOT     EQU  $2047      ;   "         "
RESET    EQU  $03F2      ;vecteur 'reset'
*
*
*        MACROS
*        ------
*
         DO   0          ;Le code n'est pas généré
*
*        Macro tabulation
*        ----------------
*        Syntaxe : TBL  Verticale,Horizontale
*
TBL      MAC             ;Nom de la macro
         PHA             ;sauvegarde accumulateur
         LDA  £.0        ;1er parametre
         JSR  TABV       ;tabulation verticale
         LDA  £.1        ;2me parametre
         STA  TABH       ;tabulation horizontale
         PLA             ;restituer accumulateur
         EOM             ;fin de macro
*
*        Macro Input
*        -----------
* syntaxe: INP  Tabul. verticale, Tabul. Horizont, Longueur entrée
*
INP      MAC             ;Nom de la macro
         PHA
         LDA  £.0        ;1er parametre
         STA  VTAB       ;tabulation verticale
         LDA  £.1        ;2me parametre
         STA  HTAB       ;tabulation horizontale
         LDA  £.2        ;3me parametre
         STA  ENT        ;fixer longueur entrée autorisée
         JSR  INPUT      ;saut au sous-programme INPUT
         PLA
         EOM             ;fin de macro
*
*        Macro afficher texte
*        --------------------
*        syntaxe: AFT   numero de message
*
AFT      MAC             ;nom de la mecro
         LDY  £.0        ;parametre
         JSR  AFFICH     ;saut au sous-programme affichge texte
         EOM             ;fin de macro
*
*        Macro afficher caractéres souris
*        --------------------
*        syntaxe: AFM   numero de message
*
AFM      MAC             ;nom de la macro
         LDY  £.0        ;parametre
         JSR  MOUSE.AFF  ;saut au SP affichage caract. souris
         EOM             ;fin de macro
*
         FIN             ;le code est de nouveau généré
*
*
*
         ORG  $4800
*
*
*        PATCH  ORG $4800
*        ----------------
*
PATCH    EQU  *
         LDA  KBD
         CMP  £$A0       ;Barre d'espace appuyée...?
         BEQ  URBUTI     ;oui
DEBUT2   EQU  *
         JSR  HOME
         LDA  CHAINE
         CMP  £$2F       ;Est-ce un préfixe...?
         BEQ  EXT.PREF   ;Oui
         JMP  BOOT       ;Non on continue
EXT.PREF EQU  *
         LDX  LCHAINE    ;Extraire préfixe
$1       LDA  CHAINE,X
         CMP  £$2F       ;Fin du préfixe
         BEQ  STORE      ;oui
         DEX
         JMP  $1
STORE    EQU  *
         LDY  £$00
$1       DEX
         LDA  CHAINE,X
         STA  BUFFER,X
         INY
         CPX  £$00
         BEQ  $2
         JMP  $1
$2       STY  LGENT
         JSR  SPREFIX    ;Etablir préfixe de boot
         JMP  BOOT       ;On redonne la main à basic.system
*
*
DEBUT1   LDA  £<DEBUT2   ;Controler la touche <reset>
         STA  RESET
         LDA  £>DEBUT2
         STA  RESET+1
         EOR  £$A5
         STA  RESET+2
         LDA  £$00
         LDX  £$FF       ;Initialisation pile
         TXS
         BIT  STROBE     ;Initialisation clavier
         JSR  COL80
         AFM  0          ;Affichage caractéres souris <1>
         AFT  0          ;Affichage textes
         AFM  1
         JSR  GPREFIX    ;préfixe courant
         AFT  1          ;
         TBL  12,18      ;Tabulation pour affichage du nom du boot
         LDX  LCHAINE    ;Récupérer la longueur
         LDY  £$00
$1       LDA  CHAINE,Y   ;Nom du boot...
         EOR  £$80       ;Affichage normal
         JSR  COUT       ;Affichage
         INY
         DEX
         BEQ  $2         ;C'est fini...?
         JMP  $1         ;Non...
$2       AFT  2          ;Affichage texte
         INP  16,13,53   ;Entrée du nouveau boot
         LDA  LGENT      ;récupérer la longueur de l'entrée
         STA  LCHAINE    ;On sauvegarde cette longueur
         LDX  £$00
$3       LDA  BUFFER,X   ;nom du nouveau boot...
         SEC
         SBC  £$80
         STA  CHAINE,X   ;...remis à sa place
         INX
         CPX  LCHAINE    ;tout est bien là...?
         BEQ  $4         ;oui
         JMP  $3         ;non
$4       LDA  £$00
         STA  CHAINE,X
         INX
         STA  CHAINE,X
         LDA  £$E3       ;Déprotéger
         STA  ACCES      ;Si protection écriture
         JSR  SETFILE
         JSR  SAVE       ;Sauver nouvelle version
         LDA  £$21       ;
         STA  ACCES      ;Protection
         JSR  SETFILE
         JSR  HOME       ;Vous partez déjà...?
         AFT  13         ;Oui, à la prochaine...
$5       JSR  BLOQ
         CMP  £$8D
         BNE  $5
         LDA  $C08B      ;Bye
         LDA  $C08B      ; "
         JMP  $BF03      ; "
         BRK
*
*
*        MODULE SAVE
*
*
SAVE     EQU  *
         JSR  MLI
         HEX  C8
         DA   OPEN.T2
         BCS  ER2
         LDA  REF.1
         STA  REF1.1
         STA  REF2.1
*
*
*        WRITE
*        -----
*
         JSR  MLI
         HEX  CB
         DA   WRITE.T2
         BCS  ER2
*
*
*        CLOSE
```

```
*       -----
*
        JSR  MLI
        HEX  CC
        DA   CLOSE.T2
        BCS  ER2
        RTS
ER2     JMP  ERREUR
*
*
*
OPEN.T2 EQU  *          ;TABLE OPEN
        HEX  03
        DA   PATH1
        HEX  0008
REF.1   HEX  00
*
*
WRITE.T2 EQU *          ;TABLE WRITE
        HEX  04
REF1.1  HEX  00
        HEX  0020
        HEX  582F
        HEX  0000
*
*
CLOSE.T2 HEX 01         ;TABLE CLOSE
REF2.1  HEX  00
*
*
*
PATH1   STR  'BASIC.SYSTEM';Nom du fichier
*
*
*       MODULE PREFIXE
*       --------------
*
*
*       SET PREFIX
*       ----------
SPREFIX EQU  *
        LDA  MEM.SD     ;Identifier drive actif
        STA  SAV.SD     ;sauvé...!
        JSR  MLI
        HEX  C6
        DA   SPREPAR
        BCS  PR.ER      ;Le préfixe est inconnu
        RTS             ;
SPREPAR EQU  *
        HEX  01
        DA   BUFFER-1
PR.ER   EQU  *
        LDA  SAV.SD     ;Récupérer drive origine
        STA  UNIT
        JSR  ONLINE     ;Retour sur drive origine
        LDX  £$FF       ;initialiser pile
        TXS
        JMP  BOOT       ;On redonne la main à basic.system
SAV.SD  DFB  00
*
*
ONLINE  JSR  MLI
        HEX  C5
        DA   T.LINE
        BCS  ER2
        RTS
T.LINE  HEX  02
UNIT    HEX  00
        HEX  0008
*
*       GET PREFIX
*       ----------
GPREFIX EQU  *          ;Gestion du préfixe
        JSR  MLI
        HEX  C7
        DA   GPREPAR
        BCS  PR.ER
        JSR  VPREFIX
        RTS
GPREPAR HEX  01
        DA   LPREBUF
*
*
VPREFIX EQU  *
        TBL  10,14
        LDX  £$00
$1      LDA  PREBUF,X   ;affichage préfixe
        EOR  £$80       ;caractère normal
        JSR  COUT
        INX
        CPX  LPREBUF    ;Affiché en totalité ?
        BEQ  $2         ;Oui...
        JMP  $1         ;Non...
$2      RTS
*
*       BUFFER PREFIXE
*       --------------
*
LPREBUF HEX  00
PREBUF  HEX  00000000000000000000000000000000
        HEX  00000000000000000000000000000000
        HEX  00000000000000000000000000000000
        HEX  00000000000000000000000000000000
        HEX  00000000000000000000000000000000
*
*
*-----------------------------------------------
*
*    ROUTINES AFFICHAGE, ERREURS & INPUT
*
*-----------------------------------------------
*
*       AFFICHAGE TEXTE
*       ---------------
AFFICH  LDA  TABLEB,Y   ;Adresse du menu (b)
        STA  TRANS
        LDA  TABLEH,Y   ;Adresse du menu (h)
        STA  TRANS+1
        LDY  £$00
$1      LDA  (TRANS),Y
        JSR  TABV       ;tabulation verticale
        INY
        LDA  (TRANS),Y
        STA  TABH       ;tabulation horizontale
$2      INY             ;caracteree suivant
        BEQ  $5         ;255 caracteres affichés
$3      LDA  (TRANS),Y
        CMP  £$8D       ;Fin de ligne ?
        BEQ  $4         ;Si oui on saute !
        CMP  £$00
        BEQ  $6
        JSR  COUT
        JMP  $2
$4      INY             ;On saute un caractere (8D)
        BEQ  $5
        JMP  $1
$5      INC  TRANS+1    ;incrementation de AH "TRANS"
        JMP  $3
$6      RTS             ;Retour au depart
TABLEB  EQU  *
        DFB  <ECR.T1,<ECR.T2,<ECR.T3,<MESER0,<MESER1,<MESER2,<MESER
                     3,<MESER4
        DFB  <MESER5,<MESER6,<MESER7,<MESER8,<MESER9,<MBYE
TABLEH  EQU  *
        DFB  >ECR.T1,>ECR.T2,>ECR.T3,>MESER0,>MESER1,>MESER2,>MESER
                     3,>MESER4
        DFB  >MESER5,>MESER6,>MESER7,>MESER8,>MESER9,>MBYE
*
*
*       AFFICHAGE CARACTERES SOURIS
*       ---------------------------
*
MOUSE.AFF EQU *
        LDA  ALTFL      ;Commutateur à <1>?
        BMI  $0         ;si oui on saute une instruction
        STA  ALTSET     ;Commutateur alt. à <1>
$0      LDA  MTAB,Y     ;Pointeur Page 0 (Poids faible)
        STA  TRANS
        LDA  MTAH,Y     ;Pointeur page 0 (Poids fort)
        STA  TRANS+1
        LDA  £$3F       ;Inverse
        STA  $32
        LDY  £$00
*
*       TABULATION
*       ----------
$1      LDA  (TRANS),Y  ;Tabulation verticale
        JSR  TABV
        NOP
        INY
        LDA  (TRANS),Y
        STA  TABH
*
*       AFFICHAGE
*       ---------
$2      INY
        BEQ  $5
$3      LDA  £$1B       ;MouseTEXTon
        JSR  COUT
        LDA  (TRANS),Y  ;Caractere a afficher
        CMP  £$8D       ;Fin de ligne ?
        BEQ  $4         ;
        CMP  £$00       ;Fin de texte ?
        BEQ  $6         ;
        JSR  COUT1      ;Affichage...
        JMP  $2
```

```
$4      INY
        BEQ  $5          ;S1 >= 255 on branche à $5
        JMP  $1
$5      INC  TRANS+1     ;Incrementation octet poids fort
        JMP  $3
$6      LDA  £$18        ;MouseTEXToff
        JSR  COUT1
        LDA  £$FF        ;Normal
        STA  $32
        RTS
MTAB    EQU  *
        DFB  <ECR.S1,<ECR.S2
MTAH    EQU  *
        DFB  >ECR.S1,>ECR.S2

*
*       MODULE INPUT
*       ------------
*
ENT     HEX  00          ;Longueur entrée
VTAB    HEX  00
HTAB    HEX  00
INPUT   EQU  *
        LDX  £$00
        LDY  HTAB
        STY  TABH        ;Tab.horiz.
        LDA  VTAB        ;Recup. tab. vert.
        JSR  TABV        ;Tab. vert.
$ENCORE LDA  £$AE
        JSR  COUT
        CPX  ENT
        BEQ  $FIN
        INX
        JMP  $ENCORE
$FIN    LDX  £$00
        LDY  HTAB
        STY  TABH        ;Tab.horiz.
        LDA  VTAB        ;Recup. tab. vert.
        JSR  TABV        ;Tab. vert.
        LDA  CURS
        JSR  COUT
BOUCLE  JSR  BLOQ
        CMP  £$8D        ;Touche "RETURN"
        BEQ  FIN
        CMP  £$FF        ;Touche "DELETE"
        BEQ  DELETE
        CMP  £$9B        ;Touche "ESCAPE"
        BEQ  SORTIE
        CMP  £$88        ;Touche "<--"
        BEQ  DELETE
        CMP  £$A1        ;Caract.<espace
        BCC  NOCHAR
        CMP  £$FE        ;Caract.>que FE
        BCS  NOCHAR
        CPX  ENT
        BCS  BOUCLE
        STY  TABH
        JSR  COUT        ;On affiche
        STA  BUFFER,X    ;On stoke A dans le buffer
        LDA  CURS        ;On reaffiche le curseur
        JSR  COUT        ;ici...
        INX
        INY
        JMP  BOUCLE
NOCHAR  EQU  *
        PHY              ;XXXX
        PHX              ;XXXX
        JSR  BIP
        PLX
        PLY
        JMP  BOUCLE
DELETE  EQU  *
        CPY  HTAB        ;Position 0 pour le curseur?
        BEQ  $1          ;si oui alors on sort
        DEY              ;On decremente la tab horiz.
        DEX
        STY  TABH        ;Le curseur recule
        LDA  CURS        ;
        JSR  COUT        ;
        LDA  £$AE        ;Un point qui efface le curseur
        JSR  COUT
$1      JMP  BOUCLE

CURS    ASC  "_"
        BRK
FIN     EQU  *
        CPX  £$03
        BCC  NOCHAR
        STX  LGENT
        STY  TABH
        LDA  £$AE
        JSR  COUT
        RTS
```

```
SORTIE  JMP  DEBUT1
*
*
*       TOUITTT...!
*       -----------
*
DUREE   DFB  $FF
HAUTEUR DFB  $FF
BIP     LDX  DUREE
$1      LDY  HAUTEUR
        BIT  HP
$2      DEY
        BNE  $2
        DEX
        DEC  HAUTEUR
        BNE  $1
        RTS
BLOQ    EQU  *           ;Attente clavier
$1      LDA  KBD
        BPL  $1
        BIT  STROBE
        RTS
LGENT   HEX  00
BUFFER  EQU  *
        HEX  00000000000000000000000000000000
        HEX  00000000000000000000000000000000
        HEX  00000000000000000000000000000000
        HEX  00000000000000000000000000000000
        HEX  00000000000000000000000000000000
        HEX  00000000000000000000000000000000
*
*
*       GESTION ERREURS
*       ---------------
*
ERREUR  EQU  *
        CLC
        CMP  £$27        ;MESER0
        BNE  $1
        LDY  £$03
        JMP  GEST
$1      CMP  £$2B        ;MESER1
        BNE  $2
        LDY  £$04
        JMP  GEST
$2      CMP  £$40        ;MESER2
        BNE  $3
        LDY  £$05
        JMP  GEST
$3      CMP  £$44        ;MESER3
        BNE  $4
        LDY  £$06
        JMP  GEST
$4      CMP  £$45        ;MESER4
        BNE  $5
        LDY  £$07
        JMP  GEST
$5      CMP  £$46        ;MESER5
        BNE  $6
        LDY  £$08
        JMP  GEST
$6      CMP  £$48        ;MESER6
        BNE  $7
        LDY  £$09
        JMP  GEST
$7      CMP  £$4E        ;MESER7
        BNE  $8
        LDY  £$0A
        JMP  GEST
$8      LDY  £$0C        ;MESER8
        PHA              ;Code erreur
        JSR  AFFICH
        LDY  £$0B
        JSR  AFFICH
        PLA
        JSR  PRBYTE      ;affichage code erreur
        JSR  BIP
$9      JSR  BLOQ
        CMP  £$9B        ;
        BEQ  $10
        JSR  BIP
        JMP  $9
$10     JMP  FERMER
GEST    EQU  *
        PHA
        JSR  AFFICH
        PLA
        JSR  BIP
        LDY  £$0C
        JSR  AFFICH
$1      JSR  BLOQ
        CMP  £$9B
        BEQ  FERMER
```

```
        JSR  BIP
        JMP  $1
FERMER  LDA  £$00
        STA  LEVEL
        JSR  MLI
        HEX  CC
        DA   T.FERMER
        BCS  CPL        ; Horreur d'erreur!!!
        LDA  £$00
        JMP  DEBUT1
T.FERMER HEX 01
        HEX  00
CPL     JMP  ERREUR     ; on ne s'en lasse pas, ça continue...
*
*
*       MESSAGES D'ERREUR
*       -----------------
*
MESER0  HEX  1419       ;3
        ASC  "   Erreur d'ENTREE/SORTIE...! "; $27
        HEX  00
MESER1  HEX  1419       ;4
        ASC  "Disque protégé en écriture...!"; $2B
        HEX  00
MESER2  HEX  1419       ;5
        ASC  "     Erreur de syntaxe...!    "; $40
        HEX  00
MESER3  HEX  1419       ;6
        ASC  "     Catalogue inconnu...!    "; $44
        HEX  00
MESER4  HEX  1419       ;7
        ASC  "   Nom de volume inconnu...!  "; $45
        HEX  00
MESER5  HEX  1419       ;8
        ASC  "      Fichier inconnu...!     "; $46
        HEX  00
MESER6  HEX  1419       ;9
        ASC  " La disquette est pleine...! "; $48
        HEX  00
MESER7  HEX  1419       ;0A
        ASC  "    Fichier verrouillé...!    "; $4E
        HEX  00
MESER8  HEX  1419       ;0B
        ASC  "          Erreur N° $"
        HEX  00
MESER9  HEX  1620       ;0C
        ASC  "Appuyez sur <ESC>."
        HEX  00
*
*
ECR.S1  HEX  011B
        ASC  'ZLLLLLLLLLLLLLLLLLLLLLL_'
        HEX  8D         ;
        HEX  021B
        ASC  'Z'
        HEX  8D         ;
        HEX  0234
        ASC  '_'
        HEX  8D         ;
        HEX  031B
        ASC  'Z'
        HEX  8D         ;
        HEX  0334
        ASC  '_'
        HEX  8D         ;
        HEX  041B
        ASC  'Z'
        HEX  8D         ;
        HEX  0434
        ASC  '_'
        HEX  8D         ;
        HEX  051B
        ASC  'Z'
        HEX  8D         ;
        HEX  0534
        ASC  '_'
        HEX  8D         ;
        HEX  061B
        ASC  'Z"
        HEX  8D         ;
        HEX  0634
        ASC  '_'
        HEX  8D
        HEX  071C
        ASC  'LLLLLLLLLLLLLLLLLLLLLL'
        HEX  00
ECR.S2  EQU  *
        HEX  0903
        ASC  'ZLLLLLLLLLLLLLLLLLLL
```

A l'essai : SuperMacroWorks

Damien Nould

SuperMacroWorks est un utilitaire destiné aux utilisateurs d'AppleWorks 1.4, leur procurant une nouvelle dimension de travail : les macro-commandes. Il s'agit de permettre l'exécution automatique de séquences de commandes par la pression d'une seule touche : les fervents de WPL apprécieront de retrouver ce qu'ils avaient douloureusement perdu en abandonnant AppleWriter. Notons en préambules que – comme toujours dans Pom's – les impressions (subjectives ?) données ici sont issues de l'usage du programme et non de celui des communiqués de presse...

SuperMacroWorks est un programme des célèbres Beagle Brothers. Il est fort heureusement complété par un mode d'emploi traduit par D. Geystor et disponible séparément.

SuperMacroWorks ne se conçoit d'ailleurs que par l'ensemble programme, mode d'emploi français et disquette 'bonus' (macros directement éditables pour en analyser le fonctionnement et éventuellement les modifier).

La mise en œuvre, sans être délicate, mérite attention : le fichier système de démarrage ne sera plus APLWORKS.SYSTEM mais SUPER.SYSTEM, le bien-nommé.

La souris

À l'usage, et peut-être même avant d'apprécier les macros-commandes, on est séduit par l'usage de la souris : cette facette justifie à elle seule l'achat ! La souris permet le choix dans les menus 'à barres', permet le déplacement dans les textes et autres feuilles de calcul, le confort est insoupçonnable. Les menus 'horizontaux' du type :

Voulez-vous vraiment faire cela ? Oui Non

sont également gérés par la souris...

Luxe supplémentaire, le défilement des textes ou de la feuille de calcul peut être continu par l'action sur le bouton de la souris.

Petit détail – pratique lorsqu'on assimile le réflexe –, en traitement de textes, le curseur souris se déplace comme à l'aide des flèches et non comme sur le Macintosh. Au bord droit, si la souris est déplacée à droite, le curseur va au premier caractère de la ligne suivante.

Macro-commandes

Pour le lecteur qui n'a jamais goûté au glossaire et au WPL d'AppleWriter, prenons l'exemple de la justification d'un texte au centre. Avec AppleWorks, il faut faire *⌘O* puis *CN* puis *Return* puis *Esc.* Grâce à SuperMacroWorks, la 'macro' C est décrite ainsi :

```
C:<awp><oa-O>CN<rtn><esc>!
```

Le langage utilisé est simple :

`C`	début de macro, on utilise C
`awp`	macro active exclusivement en traitement de textes
`oa-O`	simule pomme ouverte O
`CN`	simule la pression de CN
`rtn`	simule la pression de return
`esc`	simule la pression de escape
`!`	fin de la définition

Maintenant *⌘C* suffit à centrer le texte...

Plus subtil, l'inversion de deux mots de part et d'autre du curseur qui est exécutée instantanément par *⌘W*, car définie ainsi :

```
W:<awp><oa-left><oa-D>I<oa
-right><left><rtn><oa-righ
t><oa-right><rtn><left>!
```

Autant de commandes qui seraient fastidieuses à taper.

Il est possible d'utiliser les macro-commandes fournies directement, d'en ajouter et d'enregistrer les ajouts ou au contraire de ne concevoir une macro que pour une utilisation temporaire. Si le mot 'programme' revient souvent dans un texte, autant le mettre 'sous la pomme' en tapant :

```
⌘Wprogramme<CTRL-à>
```

Les ajouts se font en direct à tous moments ou en différé après préparation en traitement de textes.

Selon les usages, il est possible de concevoir en plus des macros fournies, des macros spécifiques réparties éventuellement dans plusieurs fichiers. À noter que lors de la préparation de macros, on peut leur affecter un usage :

- général ou ;
- traitement de textes seulement ;
- tableur seulement ;
- base de données seulement.

Pour étendre leur potentiel, les macros savent exécuter boucles et conditions (est-on bien arrivé au 'R' ? par exemple).

Un exemple ?

Parmi les macros mises à la disposition de l'utilisateur ⌘' affiche la date, mais à l'américaine :

```
Novembre 2, 1987
```

Qu'à cela ne tienne, les macros pouvant s'appeler mutuellement, il suffit de définir :

```
5:<awp><sa-'><left><left><
left><left><left><del><sa-
,><sa-w>!
```

La date, à la française, s'obtient maintenant avec ⌘5, cette macro en appelant trois autres : '⌘'', '⌘,' et '⌘w'. Bien entendu, de nombreuses autres combinaisons pourraient mener au même résultat.

Une macro est prévue pour afficher l'heure (certainement pour concurencer le programme WPL publié dans ces pages !).

Parmi les nombreuses possibilités qui, combinées, forment de véritables programmes, on note des commandes pour :

- reculer à l'espace précédent,
- rechercher la prochaine commande d'impression,
- rechercher le prochain espace,
- convertir en majuscule ou en minuscule,
- incrémenter ou décrémenter des caractères (de quoi concevoir une calculette ?),
- 'tracer' des macros pour les contrôler,
- imprimer la ligne du curseur,
- utiliser des variables,
- répéter des macros,
- attendre la saisie de texte,
- attendre une seule touche,
- afficher un message même avec des caractères souris,...

Comme avec WPL, SuperMacro-Works donne un noyau d'instructions à enrichir par compilation selon les besoins (cela fait penser au Forth) : toutes les commandes ne sont pas prévus mais comme les macros peuvent s'appeler mutuellement...

Beagle Bros Inc.
3990 Old Town ave., Suite 102C
San Diego,
CA 92110 USA
Programme : environ $50 + transport (carte de crédit)

Dimitri Geystor
Lachenaz
74350 Cruzeilles
Tél : 50 44 21 70
Mode d'emploi français, disquette 'bonus' : 65,00 F

AppleWorks & LaserJet+

Bruno Ribard

Renonçant à PostScript et aux polices de caractères intégrées, des PME optent parfois pour l'imprimante LaserJet de Hewlett Packard et non pour la LaserWriter d'Apple.

L'Apple // ne doit pas faire de complexes inutiles car il commande sans problème cette machine, comme la LaserWriter (voir n° 24 de Pom's, PostScript). Les codes de commandes indiqués ici ne concernent toutefois que la LaserJet.

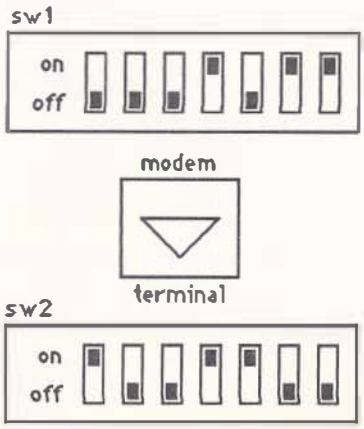
Configuration nécessaire

Pour obtenir le résultat présenté sur les exemples de la page 64, il vous faut les éléments suivants :

- Apple //e ;
- Carte Super Série Apple ;
- Câble série (analogue à celui de l'ImageWriter) ;
- Imprimante LaserJet ;
- Cartouche de caractères 92286C International1 (pour disposer des caractères accentués) ;
- AppleWorks 1.4

La carte SSC

La carte Super Série Apple doit être configurée comme pour l'ImageWriter :



Paramètres d'AppleWorks

Suivre le chemin traditionnel depuis le menu principal pour ajouter une imprimante :

Option **Autres activités**, puis **Spécifier les données Imprimante(s)**, puis **Ajouter Imprimante** puis **Mon Imprimante**.

Indiquer alors le nom que vous lui réservez, dans le cas présent : 'LaserJet+'.

Sélectionner les possibilités suivantes :

Saut de ligne après CR :	oui
Accepte saut de page :	oui
Arrêt en fin de page :	non
Largeur de chariot :	7,7 pouces
Carte d'interface :	CTRL-I 80N

Donner les codes d'impression suivants :

6 lignes/pouce :	ESC &16D
8 lignes/pouce :	ESC &18D
Début car. gras :	ESC (s3B
Fin car. gras :	ESC (s0B
Début souligné :	ESC &dD
Fin souligné :	ESC &dà
Début indice inf. :	ESC &a+37V
Fin indice inf. :	ESC &a–37V
Début indice sup. :	ESC &a–37V
Fin indice sup. :	ESC &a+37V
8 carac./pouce :	ESC (0F ESC (s0S ESC (s10H ESC &k15H
10 carac./pouce :	ESC (OF ESC (s0S ESC (s10H
11 carac./pouce :	ESC (OF ESC (s1S ESC (s10H ESC &k11H
12 carac./pouce :	ESC (OF ESC (s0S ESC (s10H ESC &k10H
15 carac./pouce :	ESC (OF ESC (s16.6H ESC &k8H
17 carac./pouce :	ESC (OF ESC (s16.6H
20 carac./pouce :	ESC (OF ESC (s16.6H ESC &k6H

Les densités 8, 10, 12 sont du type Courrier droit et autorisent le soulignement, les caractères gras et indices.

La densité 11 est du type Courrier incliné et autorise le soulignement et les indices.

Les densités 15, 17, 20 sont du type L'ine Printer droit et autorisent le soulignement et les indices.

Remarques

La configuration présenté ici n'est pas critique : un Apple //c (ou tout ordinateur avec sortie série), AppleWriter et une LaserWriter peuvent fort bien communiquer.

MultiScribe 2.0

Bernard Toméno

Dans le numéro 30 de Pom's, l'article sur Multiscribe v 2.0 faisait allusion à une prochaine version spécifique du IIGS. C'est fait !

Styleware Inc. vient de sortir Multiscribe GS 2.0 (à $99.95 si vous commandez directement : 5250 Gulfton Suite E, Houston Texas 77081, U.S.A.), tandis qu'au même moment Version-Soft nous crédite de GSWrite 2.0 (gratuit avec l'achat d'un IIGS).

Inutile de décrire ces programmes en détail : ils n'intéressent que les possesseurs d'un GS qui, "par définition" ont déjà GSWrite. Or, Multiscribe et GSWrite sont très, très, semblables : connivence ? Imitation ? Peut-être, mais plus probablement respect des 'normes Apple' par les développeurs sur ProDOS 16.

Ce ne sont toutefois pas des jumeaux univitellins, alors penchons nous sur les principales différences.

Ce qui est propre à GSWrite

- écrire en rouge et en noir (seulement...) ;
- écrire en sur-ligné rouge, vert... ;
- écrire en lettres barrées (intérêt ?) ;
- possibilités de définir et conserver plusieurs *règles de mise en page* (espacement des lignes, justifications, tabulations) et même de les baptiser du nom qu'il vous plaira. Très très commode ! ;
- marges et tabulations en centimètres et non en pouces (avec possibilités de tabulations *point par point* ou bien *décimales*) ;
- redéfinition possible de la taille des caractères (de 6 à 32 et non de 0 à 32 comme le laisse croire un des menus déroulants), ceci en plus des options standards (8 - 9 - 10 - 12 - 16 - 18) ;
- option d'annulation de la dernière saisie ;
- nombreuses possibilités de caractères "supplémentaires" par *Option + autre touche* ; on obtient une série de lettre-accent grave, lettre-tréma, lettres accolées (œ), grecques (Σ, ß, Π µ), et divers symboles.

Assez curieusement, le manuel est muet sur cette remarquable possibilité ; il signale en revanche l'astuce voyelle précédée de la touche tréma ou circonflexe *pour obtenir le caractère accentué correspondant (on peut regretter que les résultats de* Option + autre touche *soient 'mnémotechniquement' parfois peu judicieux :* Option + S *donne* ß, *tandis qu'* Option + B *donne* ' *!).*

- on peut – enfin – charger des images dans le texte (et elles y restent visibles en permanence) mais uniquement si elles sont au format GSPaint. On ne peut écrire ni à coté ni sur elles ; elles transitent obligatoirement par le presse-papier qui est de ce fait 'neutralisé'.

Ce qu'apporte Multiscribe GS

- écrire en couleur, mais pas seulement en noir ou rouge comme GSWrite : il y a une palette de 16 couleurs redéfinissables comme dans GSPaint (donc 4096 possibilités). Ceci étant dit... c'est joli à l'écran, mais il est dommage que les couleurs des rubans d'imprimantes soient encore si tristes et que l'impression matricielle soit si lente (critique valable pour les deux programmes) ;
- dessiner (mais oui), grâce à un mini-éditeur de "shapes" dans le style de GSDraw ou de TopDraw : diverses figures pleines ou vides (rectangle à bords carrés ou arrondis, cercles et ovales, lignes) avec plusieurs choix pour l'épaisseur du trait. Chaque dessin peut être indéfiniment agrandi, diminué, déplacé, superposé avec d'autres ou avec le texte : une belle réalisation ! ;
- charger des images et les mélanger avec le texte et avec les mêmes possibilités de manipulations que ci-dessus, bien qu'il ne soit pas toujours facile de respecter les couleurs de l'original qui, en revanche, peut provenir de GSPaint, DeluxePaint, PaintWork, TopDraw (quelques petites astuces à connaître, voir le manuel et un document-lettre enregistré sur la disquette) ;
- on peut, bien sûr, voir et imprimer ensemble textes et images mais, quand on dessine, on perd la vue du texte et vice versa ;
- un peu plus de 'choix standards' que dans GSWrite pour les tailles de caractères, et une sous-option permettant de faire varier la largeur des lettres de 1 point-écran ;
- 5 options de *Style* de plus que GSWrite (Ombré, Indice et Exposant, Majuscules et Minuscules) ;
- dernier atout : Multiscribe reconnaît et charge immédiatement les fichiers textes d'AppleWorks (type AWP) ;
- un regret pour finir : alphabet strictement américain : impossible d'avoir des *é è à*... même si – petits malins – vous permutez avec les polices apparemment francisées de GSWrite (compatibles, ce qui permet de les mélanger : n'en mettez toutefois pas trop... cela finit par 'déborder' en bas de l'écran).

Alors... à vous de choisir... ou attendre la prochaine version : *2.0* semble signifier *version définitive*, ce qui ne veut pas dire parfaite (en insistant bien on arrive à planter le soft américain tout autant que le français), et ce qui n'interdit pas d'espérer une version 3.0 encore plus perfonnante.

Apple // for ever

Jean-Yves Bourdin

Nouvelles du front

Les frères **Beagle**, de San Diego (Californie), plus célèbres que les frères Marx chez les Applemaniaques pour leurs programmes toujours indispensables (GPLE, Pronto DOS, Beagle Graphics...) et jamais protégés, ainsi que pour leur inénarrable sens de l'humour, ont trouvé un nouveau propriétaire. Le couple **Bert** et **Sharon Kersey**, fondateur de la société, qui en avait assez de diriger une entreprise (florissante au demeurant) plutôt que de programmer, a revendu la société... à son principal concurrent.

Il faut dire que ce "concurrent", **Mark Simonsen**, était aussi un de leurs amis, un ancien programmeur de Beagle, qui avait fondé la société **Software Touch** pour pouvoir publier des programmes également excellents, également non protégés, mais qui entraient en concurrence avec des produits Beagle. On ne peut nier par exemple que **Program Writer**, d'**Alan Bird**, surpasse et périme **GPLE**. Le même Alan Bird publiant d'ailleurs aussitôt après chez Beagle le génial **Beagle Compiler**, premier mais définitif compilateur Applesoft sous ProDOS...

La fusion de Beagle et de Software Touch ne peut être qu'une bonne nouvelle pour notre bon vieux chien, puisque Mark Simonsen a su garder et faire collaborer ces deux génies de la programmation que sont **Alan Bird** (AutoWorks) et **Randy Brandt** (SuperMacroWorks). Les premiers fruits de leur collaboration sont une série d'excellents utilitaires pour **AppleWorks** qui, à la différence des accessoires de **Pinpoint** par exemple, sont totalement intégrés à AppleWorks (voir la rubrique "le mange-disques").

L'orientation de la compagnie vers plus de produits pour les "utilisateurs" (et moins pour les bricoleurs et programmeurs) est donc confirmée. Il serait dommage cependant que Beagle renonce au Basic 16 bits et aux utilitaires ProDOS 16 que nous attendons tous.

Toujours plus

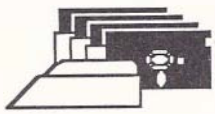
Il est inacceptable aujourd'hui de lire des rapports ou des thèses uniquement composés de ce texte sec, même pas proportionnel, que produisaient les machines à écrire de grand maman. L'informatique personnelle, c'est la publication personnelle, mise en page, variée, imagée, et même en couleurs. Sur Apple //, c'est **Newsroom**, **Certificate Maker**, ou le tout nouveau **SpringBoard Publisher**, tous de **Springboard**. Sur GS, c'est **GSWrite** de **VersionSoft** et **Graphic Writer** de **Datapak Software**, en couleurs et en super haute résolution. Pour tous, c'est aussi **Print Shop** de **Broderbund**.

Il nous faut donc des images, toujours plus d'images. Toutes les disquettes de **Clip Art** pour ces programmes sont bonnes, mais il n'y a ni ma photo ni ma collection d'estampes érotiques malgaches. La solution est connue, elle s'appelle : numérisation. Numériser une image, c'est la réduire à un nombre fini de points correspondant à la résolution graphique de votre machine et transcrire ces points en octets qui font de très belles "images-machines", images que vous pouvez ensuite afficher sur l'écran, travailler avec vos utilitaires graphiques et imprimer sur votre imprimante.

Il y a deux façons de numériser : soit à partir d'une caméra ou d'une image vidéo avec une interface spécifique (nous en parlerons un autre jour), soit... en retournant votre Imagewriter, qui, au lieu de recevoir depuis votre port série, va émettre vers lui les informations nécessaires à la reconstitution des images. Deux produits nouveaux permettent aujourd'hui aux possesseurs d'une Imagewriter de remplacer la tête d'écriture de leur imprimante par un "scanner" qui lit et transmet les images mises dans l'imprimante à la place du papier.

— **ThunderScan**, de **Thunderware** n'est plus le privilège du Mac. La version pour //e, //c ou IIGS coûte 219 dollars ;
— **Scannit**, de **Jed Design**, est offert en deux versions : **De Luxe** à 249 dollars ou le nouveau **Scannit 25** à 179 dollars.

Le mange disques



À la récente fête de l'Apple // à San Francisco, ce ne sont pas les programmes GS qui ont eu la vedette, mais la série des utilitaires AppleWorks de Beagle & Software Touch. Cette série, appelée **TimeOut**, comprend d'ores et déjà :

— un correcteur (anglais) avec un dictionnaire de 80 000 mots auquel vous pouvez ajouter le vôtre, de longueur illimitée : **TimeOut QuickSpell** ;
— un programme qui permet de transformer AppleWorks en un mini-MacWrite, d'imprimer les fichiers AppleWorks avec des fontes graphiques de toutes tailles (6 à 127 points) et de tous styles, et d'y inclure des dessins : **TimeOut Superfonts** ;
— un grapheur qui crée des graphiques directement à partir des fichiers

tableur ou base de données d'AppleWorks et les imprime : **TimeOut Graph** ;

— un nouveau programme de macros pour AppleWorks qui prend la succession de SuperMacroWorks et AutoWorks. Avec lui ApplcWorks devient un langage complet, un interpréteur de macro-commandes : **TimeOut UltraMacros** ;

— un utilitaire pour imprimer les fichiers du tableur qui dépassent la largeur de la page en les retournant à 90°, du genre de Sideways : **TimeOut SideSpead** ;

— un programme qui permet sans sortir d'AppleWorks de copier, comparer, verrouiller, déverrouiller, effacer tout fichier et de faire des copies de disque : **TimeOut FileMaster** ;

— un ensemble d'accessoires de bureau type Pinpoint ou Sidekick, permettant d'avoir instantanément accès depuis AppleWorks à une horloge, un calendrier, un agenda, une calculette, un bloc-notes, un adresseur d'enveloppes et à l'inévitable puzzle : **TimeOut DeskTools**.

Sous le capot

La meilleure carte accélérateur pour Apple //, et le seul accélérateur pour //c, viennent de sortir. Ce n'est d'ailleurs pas une carte, mais une puce qui remplace le microprocesseur et fait tourner votre machine (][,][+, //e, //c ou compatible Apple) à 4 Megaherz, 400% plus vite. **Zip Chip** de **Zip Technology** n'a pas de 'switches' (réglages au clavier), ne demande aucun slot, est garanti à vie et ne coûte que 130 dollars. Bientôt la version pour GS.

Si vous avez un nouveau //c (avec la souris en slot 7), ou un GS, ou une carte d'extension mémoire type Apple dans votre][+ ou votre //e, et que vous n'avez pas encore UN MEGA (1024 Ko) sur cette carte, vous vous privez de beaucoup de plaisirs et de possibilités. Toutes ces cartes ne demandent qu'à être remplies par des Rams parfaitement standard 41256 à 150 nanosecondes qu'on trouve chez n'importe quel marchand de composants à 30,00 F pièce maximum. Il en faut 8 pour 256Ko, 24 (mais achetez-en 25 pour la sécurité) pour "gonfler" à 1 Méga un //c ou un GS. Le seul effort demandé est celui d'ouvrir la machine (éteinte), de toucher son alimentation pour se débarrasser de l'électricité statique, et d'insérer les puces dans les supports vides qui les attendent. Pour le GS, suivre la documentation remise avec la carte d'extension mémoire (attention aux "jumpers"). Vous perdez votre garantie, et tout cela est à vos risques et périls. Mes amis et moi l'avons fait quand même.

Patchworks

Le patch suivant permet aux //c, //e, IIGS (mais pas aux II+) d'avoir sous ProDOS et Basic System 1.1 une commande CATALOG qui fait automatiquement passer en 80 colonnes (Puristes, rendez-vous au paragraphe suivant...). La commande CAT agit comme CATALOG si vous êtes en 80 colonnes, mais fonctionne comme d'habitude si vous êtes en 40 colonnes :

```
BLOADBASIC.SYSTEM,TSYS,A$2000↵
CALL-151↵
3A36 : 20 00 C3 A6 21 CA 8E N
   4323:36 N 42EF:39↵
BSAVEBASIC.SYSTEM,TSYS,A$2000↵
- BASIC.SYSTEM↵
```

Le caractère "damier" qui termine les lignes de traitement de texte dans AppleWorks 1.4 a un aspect rétro. Pour le remplacer par le moderne symbole ↵ des caractères souris :

```
BLOAD SEG.M1,T0,A$2100,L$2E00,
   B$4CDA↵
POKE 16759,205 : POKE 17163,
   205 : POKE 19451,205↵
UNLOCK SEG.M1↵
BSAVE SEG.M1,T0,A$2100,L$2E00,
   B$4CDA↵
```

Les premières versions de SuperMacroWorks de **Randy Brandt** chez Beagle Bros, programme génial de macro-commandes pour AppleWorks, avait un léger bug avec la version française 1.4 d'AppleWorks : la macro Ó^ empêche de conclure une suite de codes dans la définition de "mon imprimante". Comme cette macro n'est pas redéfinissable, il faut la changer dans le programme. Voici comment la remplacer par Ó_ :

```
BLOAD
SUPER.SYSTEM,A$2000,T$FF↵
POKE 16503, 223↵
BSAVE SUPER.SYSTEM,T$FF↵
```

Encore une partie et j'éteins

Deux nouveaux jeux marqueront l'année 87 sur Apple][par la nouveauté réelle de leur conception et le fini de leur réalisation.

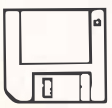
Portal d'Activision met enfin le jeu sur ordinateur à hauteur du cinéma ou de la littérature par la qualité et la quantité des émotions qu'on éprouve à y jouer. Rentré sur terre à la fin du 21ème siècle après un voyage dans les étoiles, vous y retrouvez tout... sauf les gens, qui ont disparu. Le jeu consiste à comprendre ce qui a pu se passer. Vous pouvez le faire en interrogeant Worldnet, le réseau informatique hyper-performant de cette fin de siècle, qui est encore partiellement en ordre de marche. Vous n'avez pour l'interroger qu'un vieux terminal du 20ème siècle, en deux dimensions seulement, muni d'une archaïque interface clavier, mais qui fonctionne encore (ces Apple // étaient vraiment increvables). Si vous ne lisez pas l'anglais, apprenez-le en jouant à Portal.

Accolades's Comics de **Steve Keene** et d'**Accolade** a une ambition plus modeste, mais la réalise parfaitement : celle d'être le premier jeu d'aventures et d'arcade qui soit en même temps une bande dessinée, un

Comics. Vous pilotez effectivement le jeu, feuilletez les pages, jouez à cinq jeux d'arcade différents, avec la facilité de lecture d'une BD. Si vous ne lisez pas l'anglais, vous comprendrez quand même Accolade's Comics.

On peut aussi se divertir avec des jeux plus classiques : ainsi, si quelque dinosaure de vos amis croit encore sincèrement à la fable de la prétendue "supériorité" de l'Atari sur le GS, montrez lui **Thexder** de **Sierra On Line**, le jeu d'arcade qui fait fureur au Japon, sur votre GS. Et si vous préférez les jeux à la française, essayez **Séduction 1** de **Froggy Software**, jeu de rôles qui vous met dans la peau d'un séducteur galactique.

16 bits



Ça bouge du côté du GS : **Apple USA** distribue gratuitement outre-Atlantique de nouvelles Roms et une nouvelle puce vidéo, qui éliminent pas mal de bugs de la version actuelle mais qui semble en ajouter : certains disques durs et le disque-Ram Speedisk s'en portent mal. En fait, Speedisk – par exemple – fonctionne avec les nouvelles Roms, mais pas avec la version 1.3 de ProDOS 16 livrée avec. Il convient donc dans ce cas, comme le conseillent déjà plusieurs revues américaines, de conserver la version 1.1. Apple distribue également gratuitement la dernière version du Système (version 2.0), qui occupe pratiquement toute une disquette 800Ko. Nouveautés :

- ProDOS 16 Version 1.3 (justement...) ;
- ProDOS 8 Version 1.4 (17 Avril 87) ;
- Loader Version 1.2 ;
- Launcher Version 2.0,

mais toujours Basic System Version 1.1. Sont ajoutés le Print Manager (enfin un outil pour piloter les imprimantes), des outils auxiliaires pour QuickDraw, la correction de certains bugs de la version 1.1 du système ; les outils sont présents sur la disquette sous forme compressée, et le Loader a été révisé pour permettre le chargement de ces fichiers compressés. Tout cela est à demander à votre revendeur.

SoftSwitch de **Roger Wagner Publishing**, est à la fois un 'switcher' et une 'wildcard' pour le GS. Installé comme un accessoire de bureau sous ProDOS 16, il permet de quitter une application (8 bits seulement), de la sauver en mémoire, d'en lancer une autre, de la quitter à son tour pour retrouver la précédente au point exact où vous en étiez resté, sans jamais redémarrer. Vous pouvez aussi sauver votre mémoire sur disque et retrouver des mois après le point où vous en étiez, comme avec une wildcard. Vous pouvez passer des images d'une application à une autre, etc. Il gère trois applications simultanément.

Deux points noirs, cependant, pour ce programme : d'abord, les applications concernées doivent permettre l'accès au tableau de bord, et beaucoup ne le font pas, à commencer par le Prodos 1.1.1. Ensuite, il est muni d'une protection à la fois invisible (les disques sont copiables et les copies semblent fonctionner) et vicieuse (il modifie des octets de la Ram de configuration de la machine, ceux qui sont conservés même quand on l'éteint).

Saviez-vous qu'un outil de debuging est caché dans la Rom de votre GS ? Pour avoir accès au **Memory Peeker**, tapez :

```
CALL -151↵
0 = e↵ (sortie d'émulation)
FF/1800X↵ (installation de l'outil)
⌂ - CTRL - ESC↵ (tableau de bord)
```

À lire

Apple France vient de traduire et publier un certain nombre de livres et documentations Apple. Toutes les publications techniques Apple sont une merveille de pédagogie, de clarté et d'exactitude, et sont très agréables à lire. Font donc désormais partie des indispensables à demander à votre revendeur avant qu'ils ne soient épuisés :

— **Manuel de Référence de l'Imagewriter II** (vous n'utilisez pas 10% des possibilités de cette excellente imprimante) ;
— **Le Guide des Interfaces Apple** (indispensable pour comprendre le fonctionnement de votre port série, ou choisir le bon câble pour InterPom's) ;
— **La Programmation en Basic avec ProDOS** (un des meilleurs ouvrages sur le Basic) ;
— **Manuel de Référence de l'Instant Pascal** (l'Instant Pascal est un outil extraordinaire pour apprendre et enseigner le Pascal, y compris celui du Mac. En revanche, il est inutilisable pour développer).

À lire aussi : les numéros 34 et suivants de Pom's. Vous apprendrez dans la chronique *Apple // for ever* :

- comment mettre un IBM dans votre Apple ;
- comment mettre un Apple dans votre IBM ou compatible ;
- comment utiliser les drives 800Ko compatibles Mac sur l'Apple // ;
- comment accélérer le chargement du système sur GS ;
- comment transformer un antique][+ en station Citizen Band ou en labo de physique ;
- quels langages utiliser sur GS ;
- comment piloter la LaserWriter avec un //e ou un //c ;
- ...

Accolade
20833 Stevens Creek Boulevard
Cupertino – CA 95014, USA

Activision
P.O. Box 7286 – Mountain View
CA 94039 – USA

Beagle Bros / Software Touch
3990 Old Town Avenue – Suite 102C
San Diego – CA 92110 – USA

Broderbund Software, Inc
17 Paul Drive – San Rafael
CA 94903 – USA

Datapak Software
14011 Ventura Boulevard – Suite 507
Sherman Oaks – CA 91423 – USA.

Froggy Software
33, Avenue Philippe-Auguste
75011 Paris – ☎ 43 58 25 98

Jed Design, Inc
3300 Central Avenue SE – Canton
Ohio 44707 – USA

Roger Wagner Publishing
1050 Pioneer Way – Suite P
El Cajon – CA 92020 – USA

Sierra On Line
5750 France Avenue South
Suite 123 – Edina – MN 55435 – USA

Springboard Software,Inc
7808 Creekridge Circle – Minneapolis
MN 55435 – USA

Thunderware
21 Orinda Way – Orinda – CA 94563
USA

VersionSoft
94, rue Lauriston – 75116 Paris
☎ 47 27 71 72

Zip Chip
11926 Santa Monica Boulevard
Los Angeles – CA 90025 – USA

Sur CalvaCom

Dans cette rubrique, vous trouvez un reflet de l'activité de notre boîte aux lettres (emp11) sur CalvaCom.

4) QUEST N32 - 3 I.
De: Michel LE PORT (MLP10) - 31 oct87 18h11

Pourriez-vous m'indiquer la correspondance des bornes 2 3 7 de la prise DB9 avec la prise intérieure 16 broches du Joystick ?
Merci.

Voici la correspondance des broches Joystick :

DB9	INTERNE
1	3
2	1
3	8
4	7
5	6
6	4
7	2
8	10
9	11

Ce type dinformations se trouve dans le 'Manuel de Référence Apple IIe' disponible chez tous les revendeurs Apple.

3) Pom's 22 - 3 I.
De: Arnaud KOPP (AK24) - 17 sep 87 20h56

Dans Pom's 22 page 17, vous parlez du programme RENUMBER. mais où se trouve-il?
Arnaud Kopp

Le Renumber auquel on fait allusion dans le numéro 22 de Pom's est un utilitaire qui figure sur la disquette Master DOS 3.3. Cette disquette n'est plus aujourd'hui livrée avec les ordinateurs (ProDOS oblige) mais tous les revendeurs Apple peuvent vous la fournir.

2) Pbm AuxMove - 6 I.
De: André KRZELJ (AK14) - 22 sep 87 16h46 téléphonique

Pour adapter des programmes de l'Apple // au GS, comment faire pour remplacer Aux_Move et pouvoir déplacer des blocs dans TOUTE la mémoire ?

Deux méthodes pour remplacer AuxMove :

1) Vous inspirer de la routine de VB Toméno (Pom's 32 page 10) qui est une boucle constituée de :
AF bbmmhh (LDA adresse sur 3 octets à l'envers, hh = n° page)
8F bbmmhh (STA adresse sur 3 octets à l'envers, hh = n° page).
La même méthode a été employée dans Peeks et Pokes longs (n° 30).

2) Le mieux est d'utiliser l'instruction de déplacement de blocs MVP.
MVP banc_départ,banc_destination
X contenant l'adresse sur 2 octets départ
Y contenant l'adresse sur 2 octets de destination
A contenant le nb d'octets à déplacer.
Faites PHB avant et PLB après pour préserver DB.

1) AIDE..AIDE...SVP. - 10 I.
De: Jean QUINNE (JQ13) - 07 oct 87 10h31

Bonjour !
Ici Jean Quinne de Nouvelle-Calédonie.
Bien reçu le câble pour transfert fichier entre A2E et Mac, mais rien ne marche car...
Je ne sais pas comment récupérer les fichiers de mon disque dur PROFILE qui sont en pascal 1.3 d'une part ;
je viens de recevoir la revue no 32 ; et à ma grande joie je vois

que vous décrivez à la page 63 qu'il y a une disquette de 800Ko avec un programme UFC. Est-ce que c'est la solution ou une petite solution à mon problème.
Merci et espère avoir une réponse positive...sincèrements. JEAN.

InterPom's fonctionne sous ProDOS et ne peut lire que les fichiers ProDOS. Les fichiers Pascal DOS ou CP/M doivent être convertis.
L'utilitaire Universal File Conversion est vendu par :
Quality SotfWare
21610 Lassen Street
Chatsworth
California 91311 USA
par correspondance avec carte Visa au prix de 34,95 dollars.

Cet utilitaire permet de façon simple de convertir les fichiers entre les 4 Dos : DOS 3.3, Pascal, ProDOS (ou SOS) et CP/M.

---

2) dBase- 6 l.
De: Malik AIT BACHIR (MAB10) - 19 sep 87 08h37

Bonjour
Je souhaiterais reprendre des fichiers dBase 2 crées sur Apple dans mon PCsur dBase 3. Est -ce possible avec les softs Inter-Poms pour des fichiers en CP/M.
Merci pour votre réponse.

Les fichiers dBase 2 peuvent être transmis sur l'IBM via InterPom's ; toutefois, il devront être convertis sous ProDOS avant transmission.
Bien entendu, il restera à régler les éventuelles différences de format entre dBase2 sur Apple et dBase3 sur IBM.

---

4) Problème... - 11 l.
De: Henri PIERME (HP10) - 22 sep 87 19h29

Bonjour... J'ai acheté le dernier Pom's mais j'ai qq problèmes avec le programme Lemouse... lorsqu'un menu est déroulé il reste affiché ?
Et il y a une erreur d'affichage du 2e menu que je n'ai pas réussi à corriger (c'était pire)...
Je tiens à vous préciser que je possède un Apple//e 128ko 65c02.
Merci

Effectivement, le programme LeMouse ne fonctionne que sur un //c (comme l'indique la revue). En fait, il ne fonctionne que sur //c 128Ko, ce programme étant allergique aux nouveaux 384Ko (souris en slot 7).

---

6) ROUTINE PRODOS COPIE ECRAN - 1 l.
De: Marius MARTINET (MM68) - 09 oct 87 21h54

...pouvant tourner sur cet appareil. Je suis un nouvel abonné à votre revue.

Votre message est court et nous ne sommes pas sûr de vos besoins.
Pom's a pubtié dans son numéro 17 un pgm de copie de l'écran HGR sur Epson, dans son numéro 18, une copie de l'écran DHGR sur ImageWriter et dans son numéro 19, une copie de l'écran HGR sur ImageWriter.
Dans un prochain numéro, nous publierons un pgm de copie de l'écran texte 40 ou 80 colonnes sur toutes imprimantes.

---

3) Questions - 11 l.
De: Patrice DELORME (PD28) - 11 oct 87 19h05

Bravo votre revue est fantastique.
Je possède un Apple //c, je voudrais savoir quels sont les langages
Pascal qui fonctionnent sans drive externe. En ce qui concerne Instant Pascal est-il suffisant pour développer des logiciels importants ou est il restreint à l'apprentissage du Pascal, pouvez vous me donner son prix.
Vous avez publié plusieurs montages électroniques concernant la communication (composeur de numéros, cordons minitel, détecteur de sonnerie) est-il possible de regrouper ces 3 montages en un seul afin de n'occuper qu'une seul prise de l'Apple.

- *Les Pascal d'Apple (1.2, 1.3) se contentent d'un seul lecteur, mais quel travail : il faut en permanence changer la disquette, parfois pendant l'exécution d'un programme...*
- *Instant Pascal permet l'apprentissage et le graphisme mais, étant interprété, il est plus lent et convient mal pour des applications d'importance.*

Les revendeurs Apple vous conseilleront certainement et sauront vous donner des prix.
Bien entendu, regrouper les montages est chose possible mais en tous cas, ils occuperont deux prises : le joystick et la prise communication.

---

1) Disquette //GS- 7 l.
De: Christian RICHAUME (CROO10) - 05 oct 87 19h26

J'utilise un IIGS et vos disquettes 3.5.
Pour les rendre bootables il est nécessaire de faire beaucoup plus que ce que vous conseillez : "installer les fichiers Prodos et Basic System"
Une marche à suivre très détaillée semble nécessaire, tout le monde n'ayant pas à sa disposition un informaticiende haut niveau pour aider !!
Amicalement C.R.

Et pourtant, les disquettes Pom's deviennent 'bootables' par le simple ajout de ProDOS (P8) et Basic.System. Cela suffit pour obtenir le curseur et permettre de lancer tout programme.
Bien entendu, un système réduit tel celui conseillé ne permettrait pas à GSPaint de fonctionner (il nécessite ProDOS 16) mais, comme ce n'est pas le but...
En revanche le programme QDBasic de ce numéro est le premier programme de Pom's qui nécessite ProDOS 16.

---

5) S.O.S. Applewriter II sur II GS: Possibilités... - 10 l.
De: Jean-Pierre BOULESTEIX (JPB12) - 21 oct 87 19h52

Bonjour, et merci d'avance pour le "tuyau" qui me serait très précieux :
Je travaille avec un logiciel de communication qui s'appel PROSPECTOR, après avoir "récolté" les adresses sur Minitel je dois les retravailler avec un traitement de lexte en DOS 3.3. Donc j'utilise Applewriter sur mon //e mais il m'arrive d'avoir des disquettes

140K pleines et pas lisibles dans leur totalité: la solution étant de passer mes fichiers texte sur disk 800Ko.

Donc sur II GS...? que faire, et quel serait l'autre traitement de texte DOS 3.3 qui pourrait etre accepté sur II GS? Merci d'avance, et bravo pour l'idée d'être sur Calvacom. A vous lire...

Pour utiliser un lecteur 800Ko en DOS 3.3, il faut un DOS modifié (avec UNIMATE par exemple), mais cela contrarie AppleWriter. MagicWindow conviendrait peut-être mais il n'a pas la souplesse d'AppleWriter. Convertir les fichiers en ProDOS avant de les traiter n'est pas simple non plus.

Nous ne voyons que le renoncement à Prospector ! : avez-vous essayé le programme que nous avons publié dans notre numéro 30, T.POMS.
La 'récolte' d'adresses se fait dans un fichier ProDOS non limité en taille, directement modifiable par AppleWriter (pourquoi pas avec une petite routine WPL pour accélérer les choses ?).

---

6) Conseil d'achat - 10 l.
De: David BENSIMON (DB25) - 21 oct 87 20h48

Juste un conseil à ma revue préférée :
Possesseur d'un IIe (illustre ancêtre), je bave d'envie devant le IIgs, mais voilà, le Mac + coûte actuellement le même prix que le IIgs couleur.
Conseillerez-vous à un adepte de l'assembleur du 6502, possesseur de nombreux programmes et routines... un IIgs, suite logique du IIe, ou un Mac + ?
Deuxième question : L'extension 512k du IIgs fonctionne-elle comme la carte féline ou est-elle réellement une extension mémoire (c'est-à-dire sans commutation de banc mémoire) ?
Merci d'avance.

Bonne question ! Réponse bien délicate, mais voici quelques éléments de réflexion :

Mac :
- *Rapidité du processeur —> souris plaisante à utiliser, gestion de l'écran graphique particulièrement agréable ;*
- *Ergonomie, souplesse de la gestion de disque ;*
- *Assembleur 68000 radicalement différent du 6502, c'est un autre monde ;*
- *Pas de possibilité de bâcler une petite routine sous moniteur pour un simple essai ;*
- *En Basic, on peut faire la même remarque : comment concevoir un pgm sans gérer événements, fenêtres...*

Apple IIGS :
- *Possibilité de récupérer tous les pgm Apple // ;*
- *Possibilité de travailler 'à l'Apple //' (Moniteur, Basic, Assembleur 6502...) ;*
- *Le 65816 est le prolongement du 6502, on n'est pas dépaysé, même avec 16 bits l'accumulateur, c'est bien rassurant ! ;*
- *Pas encore de vrai Basic digne de ce nom (en fait il sera bientôt disponible...) ;;*
- *Plus nous le pratiquons, plus nous lui trouvons un air Mac :*
  . *Gestion de fenêtres;*
  . *Routines graphiques QuickDraw (voir ce Pom's pour l'utilisation de QuickDraw sous Basic AppleSoft) ;*
  . *ProDOS 16 bien proche du HFS (en fait c'est le HFS qui est proche...) ;*
  . *Gestion des événements (souris, clavier, insertion disquette...) ;*
  . *Faire une application 65816 est aussi complexe qu'une application 68000 ;*
  . *Accessoires de bureau ;*
  . *AppleTalk : GSPaint permet de commande la LaserWriter... ;*
- *Lenteur du processeur, on voit les icônes se dessiner, la souris n'est pas très plaisante (quand on vient du Mac) ;*
- *Couleur (mais difficulté de programmation) ;*
- *Seuls 64Ko sont accessibles depuis le Basic Applesoft, au-delà, on commute : les 'n' mégaoctets ne sont pas en lignes mais par page' de 64Ko ;*

Dernier élément : Pom's donnera de plus en plus de pgm aux possesseurs de GS (mais sans réduire la place du Mac...)

Cruel dilemne !

---

4) RECHERHE ADRESSE... - 4 l.
De: Jean QUINNE (JQ13) - 22 oct87 10h53

Bonjour ici Jumbo Vidéo(JQ13) Quinne Jean
Connaissez-vous une adresse pour un programme d'astrologie sur Mac?
Où trouver HyperCard et MultiFinder ?
Merci et sincères salutations.

La seule adresse que nous ayons trouvée :
Decleir Louis
103/731 Bldde VCalmy
59650 Villeneuve d'Ascq
20434387
programme écrit en Mégamax C.

HyperCard et MultiFinder : disponibilité en français paraît-il en novembre/décembre chez les revendeurs Apple. Les développeurs peuvent obtenir une version US auprès d'Apple. Plus de précisions dans la cité Mac de CalvaCom...

---

4) interface utilisateur - 6 l.
De: Patrice DELORME (PD28) - 22 oct 87 19h07

Je suis en train de concevoir une interface utilisateur en LM sur Apple //c du style Macintosh mais je me heurte à des problèmes d'organisation des routines.
Pouvez-vous me conseiller un livre sur l'interface utilisateur du Mac qui débate des problèmes techniques de cet interface.
Merci d'avance.

Pour les contraintes à respecter :
Inside Macintosh vol. 1 (Macintosh User Guidelines).

Cela ne vous donnera que les principes à respecter mais aucune solution concernant la programmation de l'Apple //c : nous ne voyons pas que vous conseiller d'autre.

Bibliographie

Alexandre Duback

Excel – Bibliothèque de macros, par Mary Campbell, Que Éditions, 356 pages. Traduction.

Un livre pour ceux qui ne veulent progresser que dans le domaine des macros. Comme il s'agit d'une traduction, on a droit aux surprises habituelles : tout le premier chapitre, par exemple, indique des virgules comme séparateurs d'arguments ; la *programmation linéaire* est définie comme une programmation de haut en bas...

On peut regretter l'absence de liste récapitulative donnant une vue d'ensemble des fonctions et des commandes macros. L'intérêt principal de l'ouvrage réside dans les exemples de macros fournis.

Excel applications, par David K. Simerly, Sybex, 296 pages, 248 F.

Le prix n'est pas du tout justifié par la qualité ou l'intérêt de ce livre : ce livre fourmille d'exemples de formules mal construites et pourrait servir de base à un cours sur l'audit et l'amélioration de modèles Excel. Ainsi, de façon tout à fait originale, l'auteur invente une formule comme :

*SOMME(L9-(L9*0,4))*

là où :

*0,6*L9*

suffirait ; les exemples de cet ordre sont multiples.

L'auteur présente les codes de présentation des en-têtes et pieds de page tels qu'ils sont décrits dans la documentation d'Excel ; malheureusement, dans la réalité, ils ne fonctionnent pas comme prévu et l'auteur ne s'en est apparemment jamais rendu compte. Pour savoir comment marchent ces codes, il faut se référer au *Guide pratique d'Excel* ou à *Excel efficace*, chez Cedic-Nathan.

Graphisme et mise en page sur Macintosh, par Andréas Pfeiffer, Cedic-Nathan, 315 pages, 310 F.

Il s'agit là d'un ouvrage proche, pratiquement une seconde édition, sérieusement remaniée, du livre *Applications graphiques pour Macintosh*. L'auteur est compétent et pratique ce qu'il enseigne. Seul problème, le livre étant publié en octobre 86, il n'inclut pas Xpress et présente Ready-Set-Go dans une version 2.

L'édition sur Macintosh, par A. Chichery, J.-P. Drieux et A. Jarlaud, Dunod, 208 pages, 145 F.

Le sous-titre indique "Think Tank, More, McWrite (sic), Jazz, Word, Mac Auteur, Writer Plus". C'est tout dire, quand on sait que Jazz et Mac Auteur sont quasiment aux oubliettes, et que Writer Plus a réussi à faire regretter son retard (tant qu'il était en retard, on pouvait encore rêver...). Le Word couvert par le livre en est à la version 1.15. Certes, nous avons ce livre depuis bien deux ou trois mois mais, même à sa sortie en mars 87, il datait déjà. Si ce livre vous intéresse, attendez donc qu'il sorte en deuxième édition...

Votre permis de conduire un ordinateur, par François Gazay, Tendances et évolutions technologiques, 207 pages, 95 F.

Cet ouvrage a l'avantage d'être relativement clair et facile à lire. Les inconvénients : une présentation sobre, pour ne pas dire spartiate, et un contenu parfois un peu trop sommaire et pas tout à fait d'actualité. Ainsi, le chapitre Langages ne parle même pas d'Ada, et l'on a l'impression que l'auteur ne connaît ni les Macintosh, ni a fortiori les équipements puissants que sont aujourd'hui les 386 et autres Mac II.

Excel efficace, par Hervé Thiriez, Cedic-Nathan, 263 pages format A4, 295 F.
Disquette d'accompagnement diffusée par Éditions MEV : 175 F (800K) ou 195 F (2 x 400K).

Une centaine de thèmes, classés en 13 chapitres et lisibles séparément les uns des autres, sont traités dans cet ouvrage qui s'adresse à tous les possesseurs d'Excel ayant déjà maîtrisé le B.A.BA de cet excellent logiciel. À la fin du livre, plusieurs tables de référence vous permettent de retrouver rapidement toutes les fonctions, les fonctions macro, les commandes et les raccourcis d'Excel. Les chapitres sont les suivants : Excel sur Macintosh, L'édition, Référence aux cellules, Présentation des feuilles de calcul, Performances, Les formats, Modification des structures, L'utilisation des tableaux, Organisation des données, Les macro-fonctions, Les macro-commandes, Pot-au-feu d'astuces et Audit d'une application.
Dans la lignée de *MacAstuces* et des *Secrets de Multiplan*, ce livre est une mine de renseignements sur Excel et d'astuces d'utilisation. C'était le best-seller du stand PSI-Cedic à Apple Expo.

Attention : le bon de commande de la disquette a été omis du premier lot de livres. Sachez donc que vous pouvez commander la disquette directement auprès de Pom's (voir le bon de commande en fin de ce numéro).

Une commande a 23 heures ?

Vous êtes déjà nombreux à profiter de la messagerie CalvaCom pour vos questions techniques et à recevoir nos réponses même à des heures indues ! Votre revue est maintenant en mesure de vous proposer un nouveau service qui ajoute la rapidité télématique aux produits Pom's : la passion n'attend pas...

Si vous êtes abonnés à CalvaCom, laissez-nous vos commandes dans notre boîte à lettres (emp11) : une disquette commandée le matin avant 10 heures sera expédiée le jour même... Cette commande peut, bien sûr être saisie directement sur le clavier Minitel mais, solution plus souple, vous pourrez utiliser le fichier texte à exporter, nommé Commande.CLV sur la disquette de ce numéro.

Vous devrez nous indiquez :

- les produits souhaités,
- vos nom et adresse,
- votre numéro de Carte Bleue, Visa, Eurocard ou Mastercard,
- sa date d'expiration,
- le montant de votre commande.

En demandant un accusé de réception, vous connaîtrez l'heure de traitement...

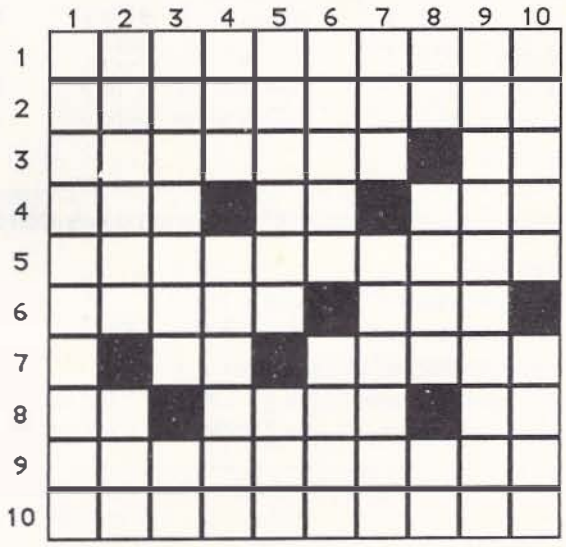
Belle occasion de s'abonner !

Mots Croisés

Gérard Vorreux, abonné du Nord, et Gérard Kittel de la région parisienne, ont été les plus prompts dans la résolution de notre problème 32 (solution ci-contre). Ils bénéficient d'un avoir de 200,00 F. Félicitations également à : Robert Baccou de Paris, Alain Pastor de Montreuil, Gisèle Perrault de Paris, Gérard Mandel de Paris, C. Jacolin de Allauch, René Martin de Cognin, M. Stankiévitch d'Aix, Ferdinand Gibert d'Eyguières, Jean Payat de Chénove, Olivier Blondeau de Saint Maur, Pascal Quilez de Bordeaux et A. Blandy. Les possesseurs de Macintosh trouveront sur la disquette d'accompagnement de ce numéro la grille et le programme de Roland Jost (application Turbo-Pascal publiée dans le numéro 32) qui remplacera avantageusement le crayon et la gomme pour la résolution du problème ci-dessous.

C	H	A	U	S	S	E	T	T	E
R	E	S	S	A	I	S	I	E	S
O	X	Y	U	R	E	S	■	O	S
C	A	M	E	■	N	U	R	S	E
O	G	P	L	A	■	Y	E	■	S
D	O	T	■	I	L	A	N	G	■
I	N	O	R	G	A	N	I	S	E
L	A	T	E	R	I	T	E	■	S
E	L	E	G	I	E	■	R	O	T
S	E	S	■	E	S	P	A	C	E

Problème 33 par Joëlle Piard



Horizontalement

1 - Tôt ou futé.
2 - Ecrit des sous-programmes, par habitude ?
3 - Assoiffer. Bien mangé.
4 - Met tout-à-fait hors de soi. Partie de charpente. Œillet sans queue.
5 - Fief.
6 - Lointains souverains. Manque d'entrain.
7 - Elcitra. Préfixe vaporeux.
8 - Imaginaires. Tout bénéfice. Drame.
9 - Premièrement, mais ce n'est pas le début.
10 - Tracassates.

Verticalement

1 - Organisme rouge. 2 - N'a que faire des paroles... - Déesse blanche.
3 - Ne supporte pas la rosière... - Personnel. 4 - Pour l'infection. Pansue.
5 - Halons. WHO pour les Anglais. 6 - Suffixe mémorable. - Henri y fit construire un château.
7 - Pour le 2.1 vertical. - Jeu d'aventures.
8 - Personnel ou personnel. Sans intérêt pour les vaches. Copule.
9 - Même pas échafaudé. 10 - Coca - Bêbêtes.

Disquettes Apple][

CLV_Pom's - 140Ko ou 800Ko	200,00 F	☐
Édite ur Vidéotex - 140Ko	200,00 F	☐
E.P.E. 5.1 - 140Ko	200,00 F	☐
E.P.E. 5.1 - 800Ko	200,00 F	☐
Max (moniteur étendu) - 140Ko	150,00 F	☐
BananaSoft - 140Ko	200,00 F	☐
Pascal - 140Ko	80,00 F	☐
Ordico - 140Ko	200,00 F	☐
Dominos - 140Ko	80,00 F	☐
COGO - 140Ko	200,00 F	☐
Ludologic - 140Ko	80,00 F	☐

Recueils Pom's

Numéro 1 (Revues 1 à 4)	140,00 F	☐
Disquettes 1 à 4	200,00 F	☐
Numéro 2 (Revues 5 à 8)	140,00 F	☐
Disquettes 5 à 8	200,00 F	☐
Numéro 3 (Revues 9 à 12)	140,00 F	☐
Disquettes 9 à 12	200,00 F	☐

_____reliures toilées
pour 6 numéros, soit un an 60,00 F ☐

Apple][800Ko 3'5 numéro 29	80,00 F	☐
Apple][800Ko 3'5 numéro 30	80,00 F	☐
Apple][800Ko 3'5 numéro 31	80,00 F	☐
Apple][800Ko 3'5 numéro 32	80,00 F	☐
Apple][800Ko 3'5 numéro 33	80,00 F	☐

Macintosh

CLV_Pom's	200,00 F	☐
Excel eff. 800K	175,00 F	☐
Excel eff. 400K	195,00 F	☐
MacAstuces	200,00 F	☐
Mac 'A'	80,00 F	☐
Mac 'B'	80,00 F	☐
Mac 'C'	80,00 F	☐
Mac 'D'	80,00 F	☐
Mac 'E'	80,00 F	☐
Mac 'F'	80,00 F	☐
Mac 'G'	80,00 F	☐
Mac 'H'	80,00 F	☐
Mac 'I'	80,00 F	☐
Mac 17	80,00 F	☐
Mac 18	80,00 F	☐
Mac 19	80,00 F	☐
Mac 20	80,00 F	☐
Mac21	80,00 F	☐
Mac 22	80,00 F	☐
Mac 23	80,00 F	☐
Mac 24	80,00 F	☐
Mac 25	80,00 F	☐
Mac 26	80,00 F	☐
Mac 27	80,00 F	☐
Mac 28	80,00 F	☐
Mac 29	80,00 F	☐
Mac 30	80,00 F	☐
Mac 31	80,00 F	☐
Mac 32	80,00 F	☐
Mac 33	80,00 F	☐

Accompagnement

A][140Ko 4	60,00 F	☐
A][140Ko 5	60,00 F	☐
A][140Ko 6	60,00 F	☐
A][140Ko 7	60,00 F	☐
A][140Ko 8	60,00 F	☐
A][140Ko 9	60,00 F	☐
A][140Ko 10	60,00 F	☐
A][140Ko 11	60,00 F	☐
A][140Ko 12	60,00 F	☐
A][140Ko 13	60,00 F	☐
A][140Ko 14	60,00 F	☐
A][140Ko 15	60,00 F	☐
A][140Ko 16	60,00 F	☐
A][140Ko 17	60,00 F	☐
A][140Ko 18	60,00 F	☐
A][140Ko 19	60,00 F	☐
A][140Ko 20	60,00 F	☐
A][140Ko 21	60,00 F	☐
A][140Ko 22	60,00 F	☐
A][140Ko 23	60,00 F	☐
A][140Ko 24	60,00 F	☐
A][140Ko 25	60,00 F	☐
A][140Ko 26	60,00 F	☐
A][140Ko 27	60,00 F	☐
A][140Ko 28	60,00 F	☐
A][140Ko 29	60,00 F	☐
A][140Ko 30	60,00 F	☐
A][140Ko 31	60,00 F	☐
A][140Ko 32	60,00 F	☐
A][140Ko 33	60,00 F	☐

Revue n° 8	35,00 F ☐	Revue n° 10	40,00 F ☐	Revue n° 11	40,00 F ☐	Revue n° 12	40,00 F ☐
Revue n° 13	40,00 F ☐	Revue n° 14	40,00 F ☐	Revue n° 15	40,00 F ☐	Revue n° 16	40,00 F ☐
Revue n° 17	40,00 F ☐	Revue n° 18	40,00 F ☐	Revue n° 19	40,00 F ☐	Revue n° 20	40,00 F ☐
Revue n° 21	40,00 F ☐	Revue n° 22	40,00 F ☐	Revue n° 23	40,00 F ☐	Revue n° 24	40,00 F ☐
Revue n° 25	40,00 F ☐	Revue n° 26	40,00 F ☐	Revue n° 27	45,00 F ☐	Revue n° 28	45,00 F ☐
Revue n° 29	45,00 F ☐	Revue n° 30	45,00 F ☐	Revue n° 31	45,00 F ☐	Revue n° 32	45,00 F ☐
Revue n° 33	45,00 F ☐						

Abonnements pour six numéros à partir du _____, à :

la revue seule	225,00 F	☐
la revue et les disquettes Apple][140Ko – 5' 1/4	525,00 F	☐
la revue et les disquettes Apple][800Ko – 3' 1/2	625,00 F	☐
la revue et les disquettes Macintosh	625,00 F	☐
la revue, les disquettes Apple][140Ko – 5' 1/4 et les disquettes Mac	925,00 F	☐
la revue, les disquettes Apple][800Ko – 3' 1/2 et les disquettes Mac	1025,00 F	☐

Envoyez ce bon et votre règlement à : **Éditions MEV – 12, rue d'Anjou – 78000 Versailles**

Nom : _____

Adresse : _____

Règlement par : CB/Visa/Eurocard/Mastercard ☐* Chèque bancaire ☐ Chèque postal ☐ Mandat ☐

numéro de la carte ____ ____ ____ ____ date d'expiration __ __ __

Montant ____, __ F Signature :

* pour les envois par avion, ajoutez 15 F par numéro et/ou par disquette soit, par exemple, 90 F pour un abonnement avec ou sans disquettes.

Câble-interface de communication Apple/Minitel

Cette liaison est indispensable pour faire fonctionner les programmes suivants :

- **MinBas** pour Apple][+, //e, //e+, //c et //GS* : programme permettant l'enregistrement des écrans Minitel, la restitution à loisir hors réseau, le stockage et/ou l'impression de copies d'écran du Minitel, et aussi l'envoi de textes ou messages sur un serveur, ces messages étant préparés à l'avance avec n'importe quel programme de traitement de textes. Programme du numéro 27 de Pom's.
- **Minitel/1** pour Macintosh : programme identique à MinBas pour Apple][, avec en plus un mini-éditeur de texte pour la préparation et le stockage des messages sans sortir du programme. Programme publié dans le numéro 27 de Pom's.
- **InterPom's 1.0** pour Applle][+, //e, //e+, //c et //GS* et Macintosh : programme de téléchargement entre Apple][et/ou Apple][et Macintosh. Ce logiciel autorise la transmission de n'importe quel type de fichier (système, texte, binaire, Basic...) en utilisant le Modem du Minitel. Programme publié dans le numéro 28 de Pom's.
- **TPom's** pour Apple][+, //e, //e+, //c et //GS** et Macintosh : programme de récupération de l'annuaire téléphonique sous la forme de fichiers texte. Numéro 30 de Pom's.
- **Ctv_Pom's** pour Apple][+, //e, //e+, //c et //GS** et Macintosh : programmes de communication pour une meilleure utilisation de CalvaCom et des serveurs 'ASCII'. Programmes présentés dans le numéro 31 de Pom's, page 2.
- **Paint → Minitel** pour Macintosh et **HGR → Minitel** pour Apple][+, //e, //e+, //c et //GS** : graphisme et Minitel, programmes proposés dans ce numéro.

* ce programme est prévu pour fonctionner avec une carte Super Série Apple ; le port série intégré de l'Apple IIGS ne convient pas. Toutefois, si vous désirez utiliser le port intégré afin d'y connecter le Minitel pour d'autres applications, commandez un câble pour Macintosh Plus.
** sur un Apple IIGS, ce programme fonctionne indifféremment avec le port série intégré ou la carte Super Série Apple. Pour connecter le port intégré du IIGS, utilisez un câble pour Macintosh Plus.

Je désire recevoir :	câble Minitel/Apple][+, //e, //e+, IIGS avec SSC	____	à 225,00 F	_____
	câble Minitel/Apple //c	____	à 225,00 F	_____
	câble Minitel/Macintosh 128 & 512Ko	____	à 225,00 F	_____
	câble Minitel/Macintosh Plus, IIGS port intégré	____	à 225,00 F	_____
	câble Minitel/IBM PC™	____	à 225,00 F	_____
	câble de liaison locale Apple II/Mac/IBM™ *	____	à 225,00 F	_____

*(préciser le type des deux machines à relier : Mac 512, Mac Plus, Apple //e,//c, IIGS, IBM PC™)
* pour les envois par avion, ajoutez 15 F par câble.

Envoyez ce bon et votre règlement à : **Éditions MEV – 12, rue d'Anjou – 78000 Versailles**

Nom : _____

Adresse : _____

Règlement par: CB/Visa/Eurocard/Mastercard ☐ Chèque bancaire ☐ Chèque postal ☐ Mandat ☐

numéro de la carte ____ ____ ____ ____ date d'expiration __ __ __

Montant ____,__F Signature :

Programme de transmission InterPom's

Ce programme décrit dans le numéro 31 vous donne la possibilité d'échanger à distance via Minitel et en local des fichiers de texte entre Apple //, Macintosh et IBM PC™. Entre deux machines de même type, tous les fichiers sont transférables : Applications, polices de caractères, accessoires de bureau, commandes, documents graphiques etc.

Je désire recevoir :	InterPom's pour Apple //, disquette 140Ko, 5'25	____	à 450,00 F	_____
	InterPom's pour Apple //, disquette 800Ko, 3'5	____	à 450,00 F	_____
	InterPom's pour Macintosh, disquette 3'5	____	à 450,00 F	_____
	InterPom's pour IBM PC™	____	à 450,00 F	_____

* pour les envois par avion, ajoutez 15 F par disquette.

Envoyez ce bon et votre règlement à : **Éditions MEV – 12, rue d'Anjou – 78000 Versailles**

Nom : _____

Adresse : _____

Règlement par: CB/Visa/Eurocard/Mastercard ☐ Chèque bancaire ☐ Chèque postal ☐ Mandat ☐

numéro de la carte ____ ____ ____ ____ date d'expiration __ __ __

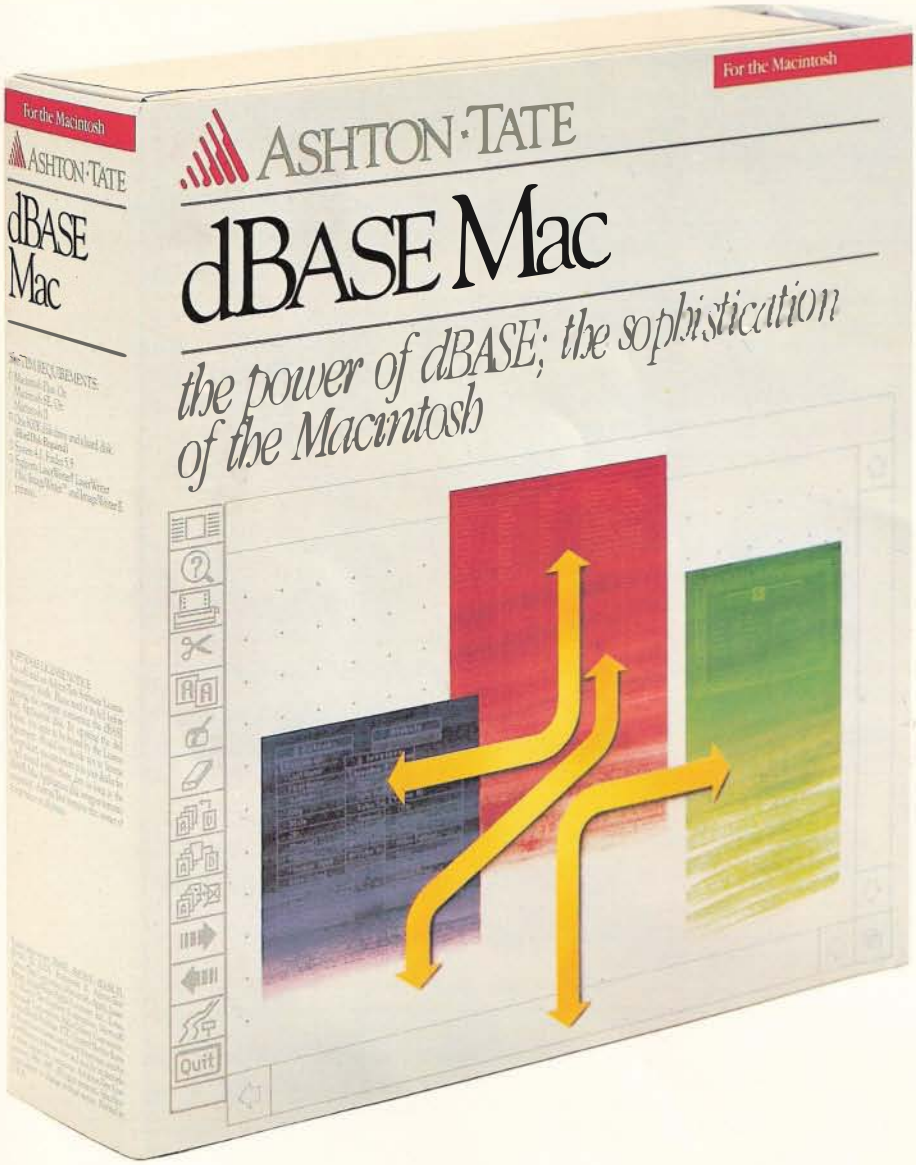
Montant ____,__F Signature :

Dépôt légal : novembre 1987, n° d'impression 789667

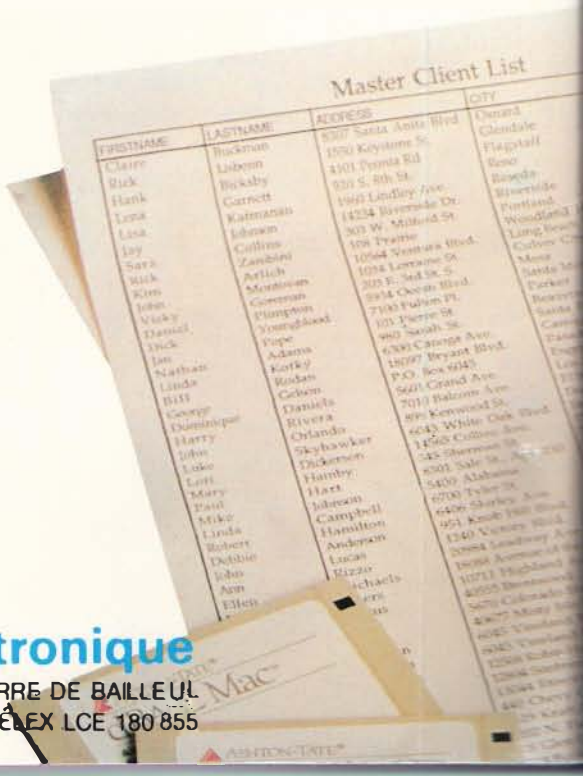
ASHTON·TATE

dBASE Mac

dBASE Mac a été spécialement conçu pour tirer parti de toutes les caractéristiques du Macintosh. Véritable gestionnaire de bases de données relationnelles, il allie toute la puissance et la fiabilité de dBASE III à la convivialité du Macintosh. Nul doute que dBASE Mac saura s'imposer comme un standard au même titre que dBASE sur PC, avec lequel il échange directement les fichiers de données. Grâce à l'interface visuelle du Macintosh, les relations multiples entre les fichiers sont représentées à l'écran. dBASE Mac possède également un puissant langage de programmation qui permet de bâtir des applications de gestion.



3 950 F (HT)



La Commande Electronique
7, RUE DES PRIAS — 27920 SAINT-PIERRE DE BAILLEUL
TÉL. 32 52 54 02 FAX N° 32 52 54 46 TÉLEX LCE 180 855